Blockchain Adoption in Organizations: Technical Barriers and Potential Solutions

Dissertation
zur Erlangung des Grades eines Doktors der Wirtschaftswissenschaft
der Rechts- und Wirtschaftswissenschaftlichen Fakultät
der Universität Bayreuth

eingereicht
von
Johannes Sedlmeir
aus
Friedberg

Dekan:	Prof. Dr. André Meyer
Erstberichterstatter:	Prof. Dr. Gilbert Fridgen
Zweitberichterstatter:	Prof. Dr. Jens Strüker
Prüfungsvorsitzender:	Prof. Dr. Daniel Baier
Tag der mündlichen Prüfung:	13.09.2023

“We tend to overestimate the effect of a technology in the short run
and underestimate the effect in the long run.”

Roy Amara (1925 – 2007)

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACL	access control list
AML	anti-money laundering
API	application programming interface
ASIC	application-specific integrated circuit
AWS	Amazon Web Services
AZ	availability zone
BFT	Byzantine fault tolerance
CFT	crash fault tolerance
CSP	cloud service provider
DAO	decentralized autonomous organization
DeFi	decentralized finance
DL	distributed ledger
DLPS	distributed ledger performance scan
DLT	distributed ledger technology
DID	decentralized identifier
DP	design principle
DSR	design science research
DPKI	decentralized public key infrastructure
eIDAS	electronic Identification, Authentication, and Trust Services
eKYC	electronic KYC
EU	European Union
EVM	Ethereum Virtual Machine
FATF	Financial Action Task Force on Money Laundering
FHE	fully homomorphic encryption
GDPR	general data protection regulation
Fabric	Hyperledger Fabric
IAM	identity and access management
IBFT	Istanbul Byzantine fault tolerance (BFT)
IdP	identity provider
IIW	Internet Identity Workshop
KYC	know your customer

MEV	maximal extractable value
MLA	Money Laundering Act
MLP	multi-level perspective
MPC	multi-party computation
NFT	non-fungible token
OLTP	online transaction processing
P2P	peer-to-peer
PBFT	practical BFT
PKI	public key infrastructure
PoET	proof of elapsed time
RBFT	redundant BFT
PKI	public key infrastructure
PoA	proof-of-authority
PoS	proof-of-stake
PoW	proof-of-work
SaaS	software as a service
SC	smart contract
SSI	self-sovereign identity
SLA	service level agreement
SLR	systematic literature review
SNARK	succinct non-interactive argument of knowledge
STS	science and technology studies
TEE	trusted execution environment
VC	verifiable credential
VP	verifiable presentation
W3C	World Wide Web Consortium
ZKP	zero-knowledge proof
zk-SNARK	zero-knowledge SNARK

Acknowledgments

I would like to express my deep appreciation and gratitude to everyone who supported me on my path to completing this doctoral thesis.

First, I would like to thank my supervisor, Prof. Dr. Gilbert Fridgen, for placing his trust in a former mathematical physicist to do project work and research in business and information systems engineering and for supporting me during the entire trajectory of my Ph.D. Your guidance and ideas influenced many of the topics in my research. I also want to thank my co-supervisor, Prof. Dr. Jens Strüker, for his steady support and inspiration. Working on projects with you was always a great pleasure.

Second, I would like to thank all my co-authors, project partners, as well as my colleagues at the FIM Research Center, the Branch Business & Information Systems Engineering of the Fraunhofer Institute for Applied Information Technology (FIT), the University of Bayreuth, the Interdisciplinary Centre for Security, Reliability and Trust at the University of Luxembourg, the University of Augsburg, the Augsburg University of Applied Sciences, and the Frankfurt University of Applied Sciences. I also want to thank my colleagues at the different businesses and public institutions I met and worked with during my Ph.D. The discussions and collaborations with all of you were very fruitful and motivated and informed my research.

Third, I would like to thank my family, in particular, my beloved partner Pia. You have always been supporting me unconditionally. Without you, I would not have come this far.

Copyright Statement

Some parts of the following sections include content taken from the published research papers included in this dissertation. To maintain the readability of the text, I sometimes omit the standard labeling of these citations.

Abstract

Blockchain technology is considered a potential solution to many challenges businesses and public institutions face within the digital transformation. Specifically, blockchain-based IT solutions can enable cross-organizational digital interactions for which no universally trusted intermediary exists for economic, political, or social reasons. Related work suggests that the technical properties of blockchain networks align very well with the requirements of many applications and emphasizes management-related adoption challenges, for instance, identifying business models or developing effective governance approaches for decentralized systems. However, when taking a closer look at blockchain projects in organizations in practice, several technical issues appear significantly more relevant than anticipated. This doctoral thesis focuses on three essential representatives of these technical adoption barriers: (1) high electricity demand, (2) low performance accompanied by high operating costs, and (3) excessive transparency. I discuss to which extent these issues are problematic when implementing blockchain technology in organizations and survey solutions already being explored in academia and practice. Furthermore, I design and evaluate novel approaches for blockchain-based IT systems. I find that sustainability issues are not problematic in general but rather an undue extrapolation from Bitcoin's high electricity needs. In contrast, performance and data visibility aspects are more fundamental and difficult to solve for organizations. According to my analysis, permissioned blockchains that have been developed to address these issues often do not provide a sufficient solution. Serverless blockchains may be useful in specific scenarios in which high performance, favorable cost stuctures, seamless integration with cloud applications, as well as fine-granular access control are crucial aspects while a high degree of centralization is acceptable. In general, blockchain-based solutions in organizations that address performance and data visibility requirements will involve complex combinations of different cryptographic components and communication layers, with verifiable computation techniques such as zk-SNARKs and the bilateral exchange of verifiable attestations according to the self-sovereign identity paradigm playing a key role.

Zusammenfassung

Blockchain-Technologie wird als eine potenzielle Lösung für viele Herausforderungen, denen Unternehmen und öffentliche Institutionen im Rahmen der digitalen Transformation begegnen, angesehen. Insbesondere können Blockchain-basierte IT-Lösungen organisationsübergreifende digitale Prozesse ermöglichen, für die aus wirtschaftlichen, politischen oder gesellschaftlichen Gründen kein allgemein vertrauenswürdiger Intermediär gefunden werden kann. Viele Arbeiten argumentieren, dass die technischen Eigenschaften von Blockchain-Netzwerken sehr gut mit den entsprechenden Anforderungen in Anwendungen übereinstimmen und betonen häufig, dass managementbezogene Herausforderungen, wie etwa die Identifizierung von Geschäftsmodellen oder die effiziente Gestaltung von dezentraler Governance, die zentralen Gründe für die langsame Adoption der Technologie sind. Bei näherer Betrachtung von Blockchain-Projekten in Organisationen zeigt sich jedoch, dass einige technische Fragestellungen wesentlich relevanter sind als erwartet. Diese Arbeit befasst sich mit drei wesentlichen dieser technischen Hürden: (1) hohem Strombedarf, (2) geringer Performanz bei gleichzeitig hohen Betriebskosten sowie (3) einem übermäßigen Grad an Transparenz. Ich diskutiere, in welchem Maße diese Hürden sich manifestieren und untersuche zahlreiche Lösungen, die bereits erforscht oder in der Praxis eingesetzt werden. Zudem stelle ich neue Ansätze für Blockchain-basierte IT-Systeme vor und evaluiere diese. Ich komme zu dem Schluss, dass Nachhaltigkeitsaspekte im Allgemeinen als wenig problematisch angesehen werden können und eher einer unzutreffenden Extrapolation des hohen Stromverbrauchs von Bitcoin entspringen. Im Gegensatz dazu sind die Performanz- und Datenschutzaspekte grundlegender und für Organisationen deutlich herausfordernder zu lösen. Meine Analyse zeigt, dass zugangsbeschränkte Blockchain-Lösungen, die gerade angesichts dieser Herausforderungen entwickelt wurden, in vielen Fällen keine ausreichende Lösung darstellen. „Serverless Blockchains" können sich in speziellen Szenarien eignen, in denen hohe Performanz, vorteilhafte Kostenstrukturen, einfache Integration mit Cloud-Anwendungen sowie feingranulares Zugangsmanagement essentielle Aspekte sind und ein vergleichsweise hoher Grad an Zentralisierung für die beteiligten Organisationen akzeptabel ist. Allgemein müssen Blockchain-basierte Lösungen in Organisationen, die typische Anforderungen hinsichtlich Performanz und Kontrolle über die Sichtbarkeit von Daten erfüllen, komplexe Kombinationen unterschiedlicher kryptographischer Komponenten und Kommunikations-Ebenen aufweisen. Techniken der „Verifiable Computation", wie etwa zk-SNARKs, sowie der bilaterale Austausch von verifizierbaren Nachweisen nach dem Paradigma selbstsouveräner Identitäten können dabei Schlüsselrollen einnehmen.

I Introduction

I.1 Blockchain – a silver bullet for information systems design?

Since the inception of the cryptocurrency Bitcoin (Nakamoto, 2008), blockchain technology as its technical backbone has raised not only consultants', founders', and established businesses' attention in various domains far beyond the financial sector. It has also inspired and attracted various research disciplines, including cryptography and computer science (Ben-Sasson et al., 2014; Butijn et al., 2020; Gudgeon et al., 2020; Zhang et al., 2019), finance and economics (Arnosti and Weinberg, 2022; Fridgen et al., 2023; Roughgarden, 2021), IT and information systems management (Goldsby and Hanisch, 2022; Hoess et al., 2023; Lichti and Tumasjan, 2023; Roth et al., 2023), and legal studies (De Filippi and Wright, 2018; Finck, 2018; Pocher et al., 2023). Blockchain is said to provide significant benefits to information systems that afford the reliable and decentralized digital exchange of information and value in sectors such as manufacturing and Industry 4.0 (Javaid et al., 2021; Leng et al., 2020), supply chain management (Dutta et al., 2020; Guggenberger et al., 2020b), trade finance (Fridgen et al., 2021b; Jensen et al., 2019), energy (Andoni et al., 2019; Roth et al., 2022b), healthcare (Aguiar et al., 2020; McGhin et al., 2019), public services and e-government (Amend et al., 2021c; Kassen, 2021; Ølnes et al., 2017), and event ticketing (Aldweesh, 2023; Regner et al., 2019), just to name a few. Consequently, research on the benefits and challenges of blockchain technology in the context of the digital transformation has been published in many consulting reports, scientific conference proceedings, and journals (Arooj et al., 2022; Dabbagh et al., 2019; Guo et al., 2021).

Academic research in information systems frequently discusses management-related aspects of adopting blockchain technology in organizations, for instance, for assessing its impact on existing business models (Weking et al., 2019), identifying new business opportunities (Lacity, 2018), integrating with the regulatory or organizational environment (De Filippi and Wright, 2018; Toufaily et al., 2021; Yeoh, 2017), or effectively forming consortia and implementing decentralized governance (Goldsby and Hanisch, 2022; Hacker et al., 2023; Liu et al., 2023). Such works often assert that from a technical perspective, blockchain brings mostly desirable properties, referring to high-level concepts like "trust" or "decentralization" and improvements in sometimes vaguely-defined dimensions like "automation" and "security". When studying materialized applications and proof-of-concept implementations that are being scaled to pilots or productive systems, issues

Paper 1–24	Verreydt et al. (2021)	Beck et al. (2017)	Egelund-Müller et al. (2017)	Hyvärinen et al. (2017)	Kranz et al. (2019)	Moyano and Ross (2017)	Nofer et al. (2017)	Notheisen et al. (2017)	Risius and Spohrer (2017)	Sedlmeir et al. (2020b)	Sunyaev et al. (2021)	Alt and Wende (2020)	Alt (2020)	Bauer et al. (2019)	Bons et al. (2020)	Drasch et al. (2020)	Hesse and Teubner (2020)	Kollmann et al. (2020)	Marella et al. (2020)	Ostern (2019)	Weking et al. (2019)	Zavolokina et al. (2020a)	Toufaily et al. (2021)	Costantinides et al. (2018)
Energy consumption	∅	∅	∅	∅	∅	∅	∅	∅	∅	✓	✓	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅	✗
Performance	✗	✗	✓	~	~	∅	~	✓	~	✓	~	~	∅	∅	∅	∅	∅	~	∅	∅	~	✗	✓	~
Transparency	✗	∅	✓	~	∅	✓	✗	~	✗	∅	✗	∅	∅	~	~	∅	∅	✗	~	~	~	~	✓	✗

Paper 25–48	Hendershott et al. (2021)	Karger (2020)	Schweizer et al. (2017)	Behnke and Janssen (2020)	Bumblauskas et al. (2020)	Janssen et al. (2020)	Kamble et al. (2020)	Upadhyay (2020)	Drummer and Neumann (2020)	Renwick and Gleasure (2020)	Sun Yin et al. (2019)	Ziolkowski et al. (2020)	Du et al. (2019)	Beck et al. (2018)	Chanson et al. (2019)	Rossi et al. (2019)	Chod et al. (2020)	Gozman et al. (2020)	Jensen et al. (2019)	Lacity (2018)	Mattke et al. (2019)	Pedersen et al. (2019)	Rieger et al. (2019)	Zavolokina et al. (2020b)
Energy consumption	∅	~	~	∅	✓	~	✗	∅	∅	∅	∅	∅	∅	~	∅	✗	∅	∅	∅	∅	∅	∅	∅	∅
Performance	~	✓	✓	✓	✓	✓	∅	✓	✓	∅	∅	✓	~	∅	✓	✓	✓	∅	∅	✓	✓	✓	∅	∅
Transparency	∅	✗	∅	~	✓	✗	✗	✗	~	✓	✓	~	✗	✓	✓	✓	~	✓	✓	~	~	~	✓	∅

Legend:

✗	Challenges or solutions discussed, but presence of misconceptions or gaps regarding challenges.
∅	Challenges or solutions not discussed.
~	Challenges or solutions mentioned but not discussed in detail.
✓	Challenges thoroughly acknowledged or solutions discussed.

Table 1: Results of the systematic literature review (sorted by corresponding journals) to which extent technical challenges to blockchain adoption for organizations are discussed in information systems research.

related to several previously less considered technical properties become apparent. Besides collecting anecdotal evidence for this perception from several industry projects in supply chain management, the energy sector, and the mobility sector, I analyze academic literature to support this perception. I first conducted a systematic literature review on publications in high-quality IS journals (including the "Basket of Eight") and the Proceedings of the International Conference on Information Systems (ICIS) related to the use of blockchain technology in organizations. In the 48 publications considered, performance issues are raised by less than 40 % of publications. Transparency issues are even only acknowledged by 25 % of publications (see Table 1). In contrast, a systematic literature study in blockchain-based supply chain management that is part of research paper 4 reveals that performance (65 %) and data visibility issues (55 %) are acknowledged much more frequently in publications that design, implement, or evaluate blockchain-based supply chain solutions – a field where blockchain adoption is arguably relatively advanced and where researchers and enterprises have implemented and tested many proof-of-concept solutions and pilot projects (e.g., Jensen et al., 2019; Mattke et al., 2019). However, requirements- and design-oriented research has thus far hardly addressed these issues, such that there is a lack of research that identifies these challenges and develops solutions based on the particular needs of organizations that want to adopt blockchain technology. This situation leaves organizations with little guidance when facing corresponding issues in their blockchain-based IT projects.

Some related work in information systems that has empirically investigated challenges of organizations when implementing blockchain-based solutions from a technical angle has already pointed out that some of the core technical properties of blockchains may indeed pose substantial challenges for adoption in organizations (Putra et al., 2023; Toufaily et al., 2021). This perspective aligns well with research from the computer science domain that heavily emphasizes potential sustainability issues of blockchains (de Vries, 2018; Goodkind et al., 2020; O'Dwyer and Malone, 2014), the challenge to scale them owing to substantial performance limitations (Ben-Sasson et al., 2019; Bonneau et al., 2015; Gervais et al., 2016), privacy and security challenges (Ben-Sasson et al., 2014; Underwood, 2016; Zhang et al., 2019), and general tradeoffs in the broad design space of blockchain-based systems with different degrees of decentralization and functionalities (Kannengießer et al., 2020; Monrat et al., 2019; Sai et al., 2021). However, while corresponding research gaps have been acknowledged also in the IS discipline (e.g., Beck et al., 2018), so far there have been few efforts to explore the implications of these technical challenges for organizations that aim to adopt blockchain-based IT products or to design innovative solutions that address these organizational requirements. The lack of decision support in problem identification and solution design for blockchain-based information systems in organizations is further aggravated by several widespread misconceptions about key technical properties of blockchain technology that may contribute to drawing the attention of decision-makers and systems engineers in the wrong direction (Ølnes, 2021). While the above-mentioned survey of 48 highly-ranked IS journal and conference publications related to blockchain technology reveals only a density of less than 10 % regarding questionable statements about energy consumption characteristics and performance aspects (e.g., regarding throughput, latency, or transaction costs), more than 20 % seem to wrongfully consider blockchains highly suitable for processing sensitive information. Considerably more publications provide no or incomplete discussions of essential technical drawbacks of blockchains when discussing the benefits of the technology for applications (e.g., 21 % and 31 %, respectively for data visibility issues). Consequently, resources may be redirected from critical aspects, such as solving data visibility issues, to less critical ones that are discussed more frequently, like energy consumption (see also Section III.2).

This doctoral thesis aims to contribute to the identification, design, and evaluation of solutions to address these said problems. It proceeds in two steps: First, I conduct an in-depth analysis of three technical challenges that are not only ubiquitous when discussing blockchain projects with decision-makers and systems engineers but also in the computer science literature: (1) high energy consumption, (2) low performance accompanied by high operating costs, and (3) an excessive degree of transparency that prohibits the implementation of effective access control and implies data visibility issues. This dissertation hence provides both information systems researchers and practitioners with guidance on the nature and significance of these technical challenges when adopting blockchain technology in their organizations. It further aims to outline how and to which extent these challenges can be addressed through the suitable choice of blockchain networks or the combination with additional, often cryptographic or game-theoretically grounded, components that take over some of the roles originally attributed to the blockchain layer. As such, this doctoral thesis builds a bridge between the business, information systems, and computer science domains by informing the discourse on implementing blockchain technology in organizations through an enhanced understanding of the severity of corresponding challenges and the key characteristics of potential solution approaches. This dissertation hence contributes to the ongoing and controversial discussion on the suitability of blockchain-based solutions to improve or disrupt information systems in the private and public sectors. While it puts a focus on investigating technical aspects, it is thoroughly informed by socio-economic and legal requirements that influence the adoption of blockchain technology adoption and its diffusion in organizations. This doctoral thesis uses methods from both the computer science and information systems domain, for instance, through both qualitative and quantitative requirements analyses and criteria- and experiment-based evaluations of proposed solution designs (see Section I.2). Specifically, it explores in depth several key technical challenges of blockchain technology when used to transform industries and identifies and evaluates different solution approaches from an interdisciplinary perspective.

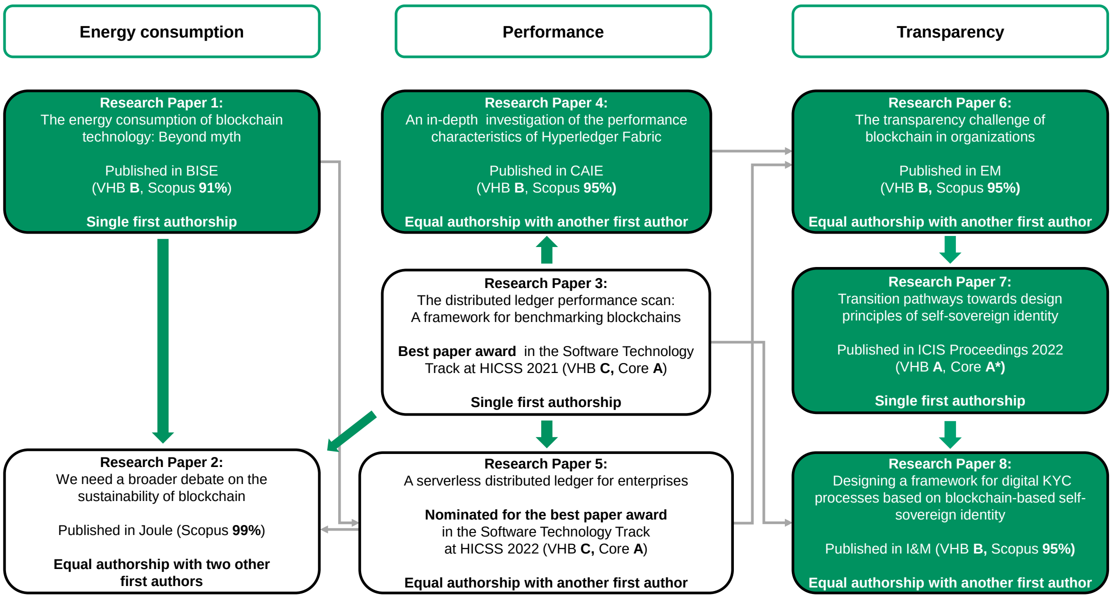

Energy consumption
Performance
Transparency
Research Paper 1:
The energy consumption of blockchain technology: Beyond myth
Published in BISE
(VHB B, Scopus 91%)
Single first authorship
Research Paper 4:
An in-depth investigation of the performance characteristics of Hyperledger Fabric
Published in CAIE
(VHB B, Scopus 95%)
Equal authorship with another first author
Research Paper 6:
The transparency challenge of blockchain in organizations
Published in EM
(VHB B, Scopus 95%)
Equal authorship with another first author
Research Paper 3:
The distributed ledger performance scan: A framework for benchmarking blockchains
Best paper award in the Software Technology Track at HICSS 2021 (VHB C, Core A)
Single first authorship
Research Paper 7:
Transition pathways towards design principles of self-sovereign identity
Published in ICIS Proceedings 2022
(VHB A, Core A*)
Single first authorship
Research Paper 2:
We need a broader debate on the sustainability of blockchain
Published in Joule (Scopus 99%)
Equal authorship with two other first authors
Research Paper 5:
A serverless distributed ledger for enterprises
Nominated for the best paper award in the Software Technology Track at HICSS 2022 (VHB C, Core A)
Equal authorship with another first author
Research Paper 8:
Designing a framework for digital KYC processes based on blockchain-based self-sovereign identity
Published in I&M (VHB B, Scopus 95%)
Equal authorship with another first author

Figure 1: Publications in this dissertation and how they relate to each other.

I.2 Structure of this doctoral thesis

This doctoral thesis comprises eight publications that contribute to identifying the challenges of blockchain adoption in organizations and developing potential solutions. Figure 1 features the embedding of these eight research papers in this dissertation and their connections through ideas, knowledge, collected data, as well as software artifacts. Green boxes represent publications in core conferences and journals in the information systems domain according to the VHB-JOURQUAL3 ranking.[1] Bold green arrows indicate that data collected or tools developed in a research project that led to the publication at the tail of the arrows also impacted or laid the foundation for the publication at the head of the arrows. Gray thin arrows indicate that motivations or ideas appearing in a publication inspired or contributed to another research paper. Figure 1 also includes information on where the research papers were published, the corresponding ranking according to VHB-JOURQUAL3, supplemented by the Scopus[2] and CORE[3] where applicable, as well as the role I took in the corresponding author teams.[4]

Each of these eight research papers is associated with one of the three core technical challenges of blockchain adoption in organizations this doctoral thesis focuses on. These chal-

[1] See https://vhbonline.org/en/vhb4you/vhb-jourqual/vhb-jourqual-3.
[2] See https://www.scopus.com/sources.
[3] See https://www.core.edu.au/conference-portal.
[4] For more details on the individual author contributions please see Appendix A.1.2.

lenges relate to the characteristics of blockchain systems in terms of *energy consumption* (research papers 1 and 2), *performance and operating costs* (research papers 3, 4, and 5), and *transparency* (research papers 6, 7, and 8). Some research papers contribute to several of these aspects. For instance, research paper 5 proposes a serverless blockchain design that mainly focuses on improving performance and operating cost aspects but also offers improvements with regard to energy consumption as well as data visibility.

To assess the significance of the individual challenges and to develop appropriate solutions, the publications in this dissertation represent a combination of several research methods from the information systems research and computer science domains. First, the publications that comprise this dissertation conduct literature reviews to identify shortcomings or acknowledged gaps in related work, such as undue generalizations of the electricity consumption of proof-of-work (PoW) blockchains to the electricity consumption of non-PoW blockchains (research papers 1 and 2), lacks of rigorous comparisons of the performance of permissioned blockchain designs that are frequently used in industry and public-sector consortia (research paper 3) and corresponding analyses of their performance characteristics in real-world scenarios (research paper 4), as well as misconceptions regarding the processing of sensitive personal or business data in blockchain-based architectures (research paper 6) and the relationship between decentralized digital identity management and blockchain technology (research papers 7 and 8). While most of these publications consider a general body of literature on blockchain-based applications, the problem statements investigated in research papers 4 and 8 focus on blockchain-based solutions for supply chain management and electronic know your customer (KYC) processes, respectively.

Second, this doctoral thesis comprises inductive reasoning, modeling techniques, and empirical data collection through experiments (quantitative) and interview studies (qualitative) to assess the severity of the identified challenges. For instance, I determine the electricity consumption of different PoW blockchain networks by deriving lower and upper bounds based on models that incorporate technical and economic boundary conditions and publicly observable quantities following de Vries (2018) and Vranken (2017) (research paper 1). Furthermore, I design and co-develop a performance benchmarking tool – the distributed ledger performance scan (DLPS) (see research paper 3) – and use it to determine key performance metrics of several permissioned blockchains commonly used in enterprise blockchain projects through a series of experiments. The design of this tool involved the identification, reviewing, and construction formal definitions of several key performance metrics, such as maximum throughput, latency, or resource utilization, as well as the conceptualization of an algorithm to determine these metrics and the implementation of a corresponding tool as a software artifact. Moreover, the DLPS automates the evaluation of large amounts of log data collected in the corresponding experiments (research paper 3). While this software artifact represents a useful contribution to the research field on its own, the DLPS also builds the basis for extended performance tests with particular permissioned blockchains, namely Hyperledger Fabric (research paper 4) and a novel "serverless" blockchain design (research paper 5), as well as for filling the literature gap on the electricity consumption of permissioned blockchains (research paper 2).

Third, I survey approaches that have been developed in related application areas of blockchains, such as cryptocurrencies and decentralized finance (DeFi), to address several technical challenges of blockchain and smart contract-based organizational information systems. With this knowledge base at hand, I study whether and how these solutions can be used to address the identified challenges. These efforts include, for instance, an investigation of the suitability of permissionless proof-of-stake (PoS) and permissioned blockchain networks for mitigating the high energy demand of PoW blockchains in research papers 1 and 2, and an evaluation of the (in-)sufficiency of permissioned enterprise blockchain designs or standard encryption techniques to thoroughly address issues related to the processing of sensitive personal or business data on blockchains (research paper 6). In this context, I also explore innovative solutions developed in startups and public blockchain ecosystems and assess the fitness of these solution approaches to address the specific organizational requirements in the blockchain-based applications under consideration by developing corresponding architectures and implementations. For example, I present how a serverless blockchain implemented by a startup can provide substantially higher throughput and more favorable cost structures for blockchains specifically during the initial phase of adoption or with high peak performance requirements (research paper 5), and argue that a revised role of blockchain in decentralized digital identity management is more suitable for the verifiable exchange of sensitive information in the context of KYC processes in banks (research paper 8).

Fourth, I evaluate the suitability of these proposed solutions to analyze their fitness and to revisit the severity of the aforementioned challenges. In particular, I quantify the electricity consumption of the less energy-demanding non-PoW-based blockchain designs and the performance characteristics and cost structure of a serverless blockchain through empirical measurements. On the other hand, I also incorporate businesses' or organizations' perspectives through qualitative analyses based on expert interviews in criteria-based evaluations of the benefits of novel approaches toward blockchain-based decentralized digital identity management and the potential role of blockchain in this context with practitioners and researchers who have deep expertise in digital identity (research paper 7) and the financial sector (research paper 8).

As such, many of the publications that are a part of this dissertation reflect core elements of the design science research (DSR) approach according to Hevner et al. (2004) and Peffers et al. (2007): Starting from a literature gap or business need with high practical relevance and systematic and rigorous analyses of requirements or challenges, as well as existing design knowledge from recent advancements in computer science, I propose novel blockchain-based solutions and evaluate them involving the current knowledge base and analyses of qualitative and quantitative data. In doing so, I also elevate the design knowledge I gained during the research process to generalizable guidelines, for instance, on the role of blockchain for decentralized digital identity management (research paper 8). While only research papers 7 and 8 formally cover the full DSR process, also the combination of research papers 4 and 5 can be considered DSR, as they provide a combination of extensive problem identification (an in-depth study of performance issues among one of the arguably most sophisticated permissioned blockchain solution, i.e., Hyperledger Fabric (Androulaki et al., 2018)) and a criteria-based evaluation of how a novel serverless blockchain design can address these challenges (with a focus on performance and cost aspects). In all these efforts, the DLPS developed in research paper 3 provides a crucial means to benchmark implemented solutions and to provide empirical data for discussions on energy consumption (research paper 2) and performance (research papers 4, 5, and 8). Moreover, key concepts from computer science, particularly distributed systems and cryptography, are included in both the identification of challenges and the design and evaluation of solutions. As such, corresponding underlying theories, such as on the construction of general-purpose verifiable computation through succinct non-interactive arguments of knowledge (SNARKs) and in particular zk-SNARKs to improve on performance and data minimization, can be considered a kernel theory from the natural sciences that heavily influences my design research processes and corresponding artifacts (Gregor and Hevner, 2013; March and Smith, 1995). This connection once more emphasizes the nature of this dissertation as an interdisciplinary work that aims to bridge the information systems and computer science domains in studying technical challenges encountered in the organizational adoption of blockchain technology.

As discussed above, this doctoral thesis focuses on three key areas of technical challenges related to blockchain networks: electricity consumption, performance and costs, and excessive transparency. Before taking a more general perspective that incorporates related work on technical challenges of blockchain-based information systems from a broader perspective, I present these three streams with the research papers they represent individually.

Stream 1: Energy consumption

Research paper 1 surveys and extends the scientific body of knowledge on the energy consumption of blockchains, which is mainly driven by the electricity needs of Bitcoin. While the enormous electricity consumption of PoW-based cryptocurrencies is essentially uncontested, research paper 1 identifies and counters several ubiquitous misconceptions in the broader area. For instance, the electricity consumption of PoW blockchains is often wrongly considered to grow linearly with the number or complexity of processed transactions. Moreover, unlike often stated, improving hardware efficiency does not have a lasting impact on the energy consumption of PoW blockchains. Another core contribution of research paper 1 is the extension of the discussion of electricity consumption aspects beyond PoW cryptocurrencies, including PoS and permissioned (enterprise) blockchains. Research paper 1 argues that the energy needs of these blockchain networks are orders of magnitude lower than those of PoW blockchains and that the heterogeneity of blockchain technology implies that the high electricity demands of PoW cryptocurrencies like Bitcoin should not be interpolated to blockchain technology as a whole. Research paper 1 also determines estimates for PoW blockchains beyond Bitcoin and emphasizes that the energy needs of PoW blockchains are predominately determined by economic parameters (e.g., cryptocurrency prices, the number of new coins generated per time to reward miners, cumulative transaction fees, and electricity prices), not technical ones.

Independently of my own work, analyses of the energy consumption of several PoW blockchains beyond Bitcoin were published by Gallersdörfer et al. (2020). Later, detailed analyses of the energy consumption of permissionless non-PoW blockchains and in particular PoS-blockchains followed (e.g., Gallersdörfer et al., 2022). However, quantitative estimates of the electricity consumption of permissioned blockchains widely used in enterprises were still not available. I hence used the DLPS (see research paper 3) for the extensive measurement of the electricity consumption of different enterprise blockchains, substantiating the estimates originally given in research paper 1. Through this complete picture, research paper 2 argues that while the high electricity consumption of PoW blockchain network remains problematic, non-PoW blockchains and in particular permissionless PoS blockchains and the permissioned blockchains with voting-based consensus mechanisms that are frequently used in enterprises have such low energy needs that corresponding blockchain-based IT solutions even have the potential to achieve net energy and carbon emission savings and, therefore, to contribute to sustainability if they can facilitate the improvement or digitalization of additional analog processes. This research stream hence suggests that when adopting non-PoW blockchains, energy consumption can hardly be regarded as problematic from the perspective of organizations, at least when taking a scientific perspective instead of considering undifferentiated public opinion.

Stream 2: Performance

The second stream of research in this dissertation is motivated by the often substantial performance requirements of information systems in organizations. For this reason, organizations have explored dedicated permissioned blockchain designs that restrict participation and, therefore, allow for higher throughput, lower latency, and a higher degree of accountability (see also Section II). Yet, the replicated processing of information makes also permissioned blockchains significantly slower than centralized systems. Because of a lack of scientifically validated blockchain performance testing frameworks, I co-developed the DLPS in research paper 3 in collaboration with a car manufacturer that needed to assess the suitability of different enterprise blockchains for a blockchain-based supply chain management implementation. An initial performance analysis of nine different permissioned blockchains networks with the DLPS yielded two key observations: First, Hyperledger Fabric can be considered one of the permissioned blockchain frameworks with the highest performance, although like-to-like comparisons are difficult to achieve. Second, the limited maximum throughput of Hyperledger Fabric even under highly favorable conditions (e.g., a small number of nodes, simple transactions, and low network latencies) suggests that performance may indeed become a bottleneck in production-grade, scaled organizational deployments.

From this starting point, I first investigated in more depth the performance characteristics of Hyperledger Fabric by surveying different design parameters that practical deployments exhibit or need to define, and by conducting corresponding sensitivity analyses regarding different performance metrics (research paper 4). Indeed, I found that while the effect of intercontinental network latencies and computationally-intensive tasks on throughput and latency is moderate, some configurations, such as large network sizes and data-heavy transactions, can significantly reduce the maximum throughput, sometimes to only a double- or single-digit number of transactions per second. This observation suggests that for applications such as blockchain-based intercontinental supply chain management, performance indeed needs to be considered from the ground and that either substantial improvements on the blockchain base layer or modifications of the workload that needs to be processed on the blockchain layer are necessary. Moreover, the analysis indicates that increasing the computational capacity of the participating nodes can help increase throughput, but only to a limited extent and potentially involving disproportionately high costs.

The second direction I investigated from this starting point is not only motivated by these significant performance limitations, particularly when it comes to applications such as cross-organizational data lakes, but also by several further technical challenges that emerge when using enterprise blockchains in organizations, for instance, from an integration point of view. Research paper 5 systematically collects these challenges of both permissionless and permissioned blockchain use in organizations and discusses how "serverless blockchains" that construct an (otherwise server-based) node with cloud-native and elastically scalable code execution as-a-service components ("serverless computing") can address them. Research paper 5 finds that while this design further centralizes the operation of the blockchain even when compared to permissioned blockchains, this tradeoff may be acceptable for organizations in many scenarios: On the one hand, the serverless blockchain design can indeed address many of the most pressing technical challenges associated with blockchain adoption in organizations; particularly improving dramatically on throughput and cost structure. On the other hand, existing contractual agreements between these organizations and their cloud service providers (CSPs) may make the higher degree of centralization acceptable as long as every organization is free to pick the CSP of their choice.

Stream 3: Transparency

Related work often considers transparency as a purely beneficial property of blockchains (e.g., Bons et al., 2020; Centobelli et al., 2022), as it is said to facilitate trust. However, information systems research on the compliance of blockchain solutions with the general data protection regulation (GDPR) (e.g., Rieger et al. (2019)) and a basket of seminal publications from the computer science domain (e.g., Kannengießer et al. (2020) and Zhang et al. (2019)) emphasize that the symmetric data visibility for all participants in a blockchain network – a direct consequence of the replicated processing of information on blockchains – is often problematic. Indeed, my research on performance aspects in the context of the automotive supply chain indicates that blockchain-based information systems in organizations require additional capabilities for access control (see research paper 4). Also the survey of companies trying to implement or adopt blockchain-based solutions in research paper 5 suggests that fine-granular access control is often an indispensable aspect of enterprise information systems, particularly when multiple competing organizations are involved. Yet, this is precisely the scenario where blockchain technology is generally considered most beneficial (Bauer et al., 2022; Fridgen et al., 2018c). Consequently, I discuss the two-sided sword of transparency exploratively in research paper 6 and link it to the discussion on the adoption of blockchain technology in organizations. Research paper 6 emphasizes the necessary distinction between transparency and verifiability, structures the types of sensitive data involved in different application areas for which blockchain technology is considered promising, and identifies solution approaches to keep this data confidential despite using it in blockchain-based information processing for enhancing verifiability. These solution approaches include secure computing techniques, including secure hardware and cryptographic alternatives such as fully homomorphic encryption (FHE), multi-party computation (MPC), and zero-knowledge proofs (ZKPs). Another approach involves the bilateral (“off-chain”) verifiable exchange of sensitive information, such as personal or business master data, using decentralized digital identity management. As the adoption of secure computation can still be considered relatively complex and has only happened in a few selected blockchain-based applications in practice thus far (e.g., Mattke et al., 2019), I conducted empirical research on these decentralized or self-sovereign identity (SSI) management systems to better understand the connection between blockchain technology and the handling of verifiable, sensitive (in particular personal) information. Research paper 7 builds a foundational understanding of the key design principles of this SSI paradigm when applied in regulated, organizational environments. It finds that while blockchain technology itself has often been considered as an enabler and core building block of decentralized digital identities (e.g., Mühle et al., 2018; van Bokkem et al., 2019), this perspective should be challenged. Research paper 8 then investigates how moving sensitive personal information off the blockchain layer into bilateral interactions between SSI wallets can resolve not only privacy but also scalability issues of blockchain-based, decentralized information systems that have been proposed before. Research paper 8 thus finds that the role of blockchain for both digital identity management per se and for information systems that manage verifiable digital identity attributes has likely been overestimated.

The following doctoral thesis presents these eight papers in the context of related work and crystallizes a broader picture of the key body of academic knowledge on the technical challenges of blockchain adoption and corresponding solution approaches. To this end, Section II first introduces the key technical characteristics of permissioned and permissionless blockchains that are responsible for the technical challenges discussed in this dissertation. Section III then discusses various technical challenges of blockchain-based information systems as pointed out by related work and the publications that comprise this doctoral thesis, as well as corresponding solutions. After an initial broad overview of technical challenges that also cover several security aspects in Section III.1, I discuss particularly electricity consumption (Section III.2), performance (Section III.3), and transparency issues (Section III.4) by refining related work on these aspects through empirical analyses and by surveying and assessing different solution approaches. Section IV concludes this dissertation. It summarizes the key contributions and findings of this doctoral thesis, lists corresponding limitations, and outlines avenues for future research. I conclude with an acknowledgment of related work in the research group I was part of, as these publications represent an invaluable knowledge base underlying this dissertation (see Section V). Section VI provides the bibliography for this dissertation. Appendix A lists my individual contributions to the publications that are a part of this dissertation, as well as additional publications I co-authored during my Ph.D. that are not formally part of this doctoral thesis. The appendix also provides the full research papers associated with the publications included in this dissertation.[5]

[5] I confirm that I hold the right to include the full publications in this dissertation. Research papers 1 and 6 have been published open access (CC BY 4.0) as they are covered by the *Projekt DEAL*. Research papers 3, 5, and 7 represent conference proceedings (co-)organized by the Association for Information Systems (AIS) where the copyright has remained with the authors. Lastly, Cell Press and Elsevier permit the publishing of research papers in a doctoral thesis as long as the formatting is not adjusted to the final layout in the journal publication, such that also research papers 2, 4, and 8 are covered.

II A short technical history of blockchain technology

At its core, blockchain technology facilitates distributed information systems characterized by synchronized, transparent, and tamper-proof record-keeping without the need for intermediaries or dedicated centralized authorities (Butijn et al., 2020) like a centralized platform provider (Catalini and Gans, 2020). Already before Nakamoto (2008)'s seminal paper "*Bitcoin: A peer-to-peer electronic cash system*" that introduced blockchain technology, fault-tolerant computing systems that removed the need for trusting and depending on a distinguished intermediary were well-known in computer science as "replicated state machines" (Buchman, 2016; Ongaro and Ousterhout, 2014; Vukolić, 2016). Both traditional replicated state machines and the blockchain networks underlying modern cryptocurrencies involve multiple computers ("nodes"), each maintaining an identical copy of the "state" – a database with well-defined rules that determine which updates ("state transitions") result from the processing of transactions. Nodes communicate with each other in a peer-to-peer (P2P) network to achieve agreement ("consensus") on a selection and order of transactions to be applied. Each node then individually applies these transactions to its local state. If transactions are deterministic, this construction ensures the (eventually) same state ("synchronization") among all nodes that act honestly and do not crash, even in the presence of a certain threshold of "faulty" nodes that do act maliciously or crash (Bagaria et al., 2019; Butijn et al., 2020; Schneider, 1990).

Blockchain technology as introduced by Nakamoto (2008) and subsequently adapted by many researchers allows extending the effective construction of replicated state machines from closed, "permissioned" systems in which only pre-registered and, therefore, accountable entities (Beck et al., 2018) can participate (e.g., through mutually positive-listing the IP addresses and cryptographic public keys of their nodes) to open, "permissionless" systems through a few crucial changes: (1) adding Sybil resistant mechanisms (Biryukov and Feher, 2020; Platt and McBurney, 2023) to ensure that despite the pseudonymity in a permissionless system, malicious actors cannot achieve excessive degrees of control by introducing many bogus accounts at negligible costs, (2) non-determinism in transaction selection and ordering (Kolb et al., 2020; Sankagiri et al., 2021) to address a fundamental tradeoff between consistency ("safety") and availability ("liveness") guarantees that holds for previous deterministic constructions of replicated state machines according to the FLP theorem (Fischer et al., 1985), (3) efficiency improvements by distributing transactions in batches ("blocks") and storing them in an append-only list ("ledger") and (4) the use of authenticity-enhancing cryptographic primitives beyond digital signatures for authorizing transactions, specifically hash-pointers and related constructions, such as Merkle trees or Merkle Patricia tries, to simplify the verification of the correctness of the state of a node, particularly for nodes that join the system (research paper 5; Garrido et al., 2022; Ruan et al., 2020). These modifications contribute to facilitating a stronger degree of decentralization and, therefore, higher integrity guarantees of blockchains compared to earlier permissioned replicated state machine designs (Zhang et al., 2019).

As indicated in Section I.1, and as I will discuss in more detail in Section III, some permissionless blockchains exhibit a particularly high electricity consumption, and especially those with an intended high degree of decentralization face aggravated issues regarding performance and operating costs as well as an often excessive degree of transparency. Consequently, the high hopes for blockchain-enabled decentralized information systems led to the unprecedented popularity also of less decentralized permissioned blockchain designs, which can be considered only minor modifications (or remakes) of early replicated state machine designs that do not provide Sybil resistance (Angelis et al., 2017; Kolb et al., 2020). These systems employ what is often called "voting-based" consensus mechanisms. Examples comprise crash fault tolerance (CFT) consensus algorithms, such as Clique or RAFT ("reliable, replicated, redundant, and fault-tolerant") (Angelis et al., 2017; Ongaro and Ousterhout, 2014), or BFT consensus mechanisms, such as practical BFT (PBFT) (Castro, Liskov, et al., 1999) or HotStuff (Yin et al., 2019). These consensus algorithms rely on only a relatively small number of select nodes to validate transactions and reach consensus. In many of these consensus algorithms, a leader is democratically chosen among the nodes ("one node, one vote") in every time period ("epoch"). The leader is then responsible for proposing one or multiple blocks within this epoch (Buchman, 2016). The other nodes in the blockchain network validate a proposed block and either vote for or against it, depending on whether all transactions within the block are legitimate in the specified order. If the majority (often a threshold of at least $\frac{1}{2}$ or $\frac{2}{3}$) of nodes agrees, all non-faulty nodes add the block to their local ledger and apply it to their local state. To avoid inconsistencies that may arise from specific faults, for instance, when the current leader crashes after sending a new block only to a proper subset of nodes in the network or when a malicious leader intentionally sends different blocks to different nodes, these protocols usually make use of digitally signed and, thus, accountable communication and employ two (CFT) or three (tolerance against arbitrary faults, BFT) rounds of incrementally affirmative messages (Angelis et al., 2017; Castro, Liskov, et al., 1999). As the number and identity of nodes are known these consensus protocols can proceed to the next step whenever a specific threshold of nodes has responded (Lamport et al., 1982). This design implies that the inclusion of a certain transaction in the blockchain can quickly be finalized, i.e., the transactions cannot be reverted anymore. Consequently, these consensus algorithms typically exhibit lower time to block confirmation ("latency") than can be achieved in permissionless blockchains (research paper 1; Gervais et al., 2016).

Alternative consensus mechanisms are required for permissionless blockchain networks, where any entity can participate with one or several nodes: As there is no predefined list of nodes, a "one node, one vote" approach is not suitable (research paper 1; Platt and McBurney, 2021). Permissionless consensus mechanisms are designed to handle an indefinite number of nodes to validate transactions and reach consensus. They do so by combining a Sybil resistance mechanism, such as PoW or PoS, with a decision rule that allows identifying the canonical sequence of blocks among potentially many alternatives.

Sybil resistance mechanisms couple the weight of a node in decision-making linearly to a scarce resource that cannot be replicated at negligible costs (research paper 1). Arguably the most popular Sybil resistance mechanisms are PoW and PoS. In PoW, this scarce resource is computational power, which is dependent on the availability of compute hardware and electricity and, ultimately, capital (Arnosti and Weinberg, 2022; Budish, 2018). To achieve that this linear coupling is verifiable in the decentralized P2P network, nodes compete to solve a cryptographic puzzle for which random trial-and-error is acknowledged to be the best strategy and for which the probability of finding a solution within a certain time period is proportional to the amount of computational resources invested (Eyal and Sirer, 2014; Nakamoto, 2008). The first node to solve this cryptographic puzzle may propose a new block and is rewarded with newly created cryptocurrency coins. In contrast, PoS-based consensus selects block proposers or validators according to the amount of cryptocurrency they hold as collateral, i.e., the weight of a node in decision-making is linearly coupled to their capital (Roşu and Saleh, 2021; Saleh, 2021).

Decision rules can be classified into longest chain rules ("Nakamoto consensus") that achieve only probabilistic finality[6] and a family of protocols that first select a subset of nodes through a Sybil resistant mechanism (e.g., PoW or PoS) and then conduct a second round of voting-based (e.g., BFT) consensus that immediately finalizes blocks (Bagaria et al., 2022). Permissionless consensus mechanisms typically exhibit higher latency than permissioned consensus algorithms, as the number of nodes involved in the (first step of) the consensus protocol is unknown. As lower latency also facilitates a more uniform dis-

[6] However, the probability of a transaction being reverted decreases exponentially with time.

tribution of resource utilization in terms of bandwidth, CPU, and memory, permissionless blockchains tend to provide lower performance than permissioned blockchains under the same resource requirements, at least when storage is not the bottleneck (see the discussion in Section III).

In Section III, we will see that the choice of Sybil resistance mechanism has a profound impact on the sustainability aspects of permissionless blockchains, whereas the impact of the consensus rule on energy consumption is limited. Moreover, while the impact of both the Sybil resistance mechanism and decision rule on performance and scalability aspects is small, the Sybil resistance mechanism seems to be more relevant also for these dimensions. Consequently, in the following, we will follow a common terminology and consider PoW and PoS as consensus mechanisms, while they, in fact, could represent a family of consensus mechanisms with different decision rules.

In summary, while both replicated state machines and blockchain technology facilitate fault-tolerant distributed systems, blockchain technology introduces additional mechanisms to facilitate open and, thus, more decentralized systems. On the other hand, permissioned blockchain designs also provide more means to control the degree of decentralization. Both permissionless and permissioned blockchains are characterized by fully replicated information processing through maintaining an append-only list of transactions ("ledger") and a synchronized state as a result of locally storing and executing deterministic transactions (Butijn et al., 2020; Kolb et al., 2020).

III Technical challenges and potential solutions

III.1 General overview

Related work particularly from the computer science domain has identified and discussed a broad set of technical challenges of blockchain-based information systems. These challenges can be associated with the different layers of blockchain-based applications.

First, as blockchain technology itself builds on multiple layers, including physical computing and networking devices as well as software components (e.g., the implementation of consensus and execution clients), problems in either of these layers can result in vulnerabilities (Sai et al., 2021). While blockchains are in principle designed to cope with a certain threshold of faulty nodes, centralization can lead to a lower acceptable number of faults, as failures are correlated and, thus, the total number of nodes providing fault tolerance is effectively reduced. With such issues often only being detectable in the case of a concrete incident, centralization can even be considered a systemic risk, as defined, for instance, in the financial sector (Haldane and May, 2011) or in power systems (Körner et al., 2022). Consequently, even though common constructions of permissioned (e.g., Castro, Liskov, et al., 1999) and permissionless PoW (e.g., Dembo et al., 2020) and PoS (e.g., Kiayias et al., 2017) blockchain networks can formally be proven secure under reasonable assumptions, the presence of vulnerabilities that affect many blockchain nodes at the same time may pose a considerable threat to the consistency and availability guarantees of blockchain networks in practice. For instance, most Ethereum nodes operate on Linux servers running on silicon manufactured by Intel, so that a single rogue organization or a single compromised dependency in the Linux kernel or libraries commonly used in UNIX-based systems (e.g., LibP2P or OpenSSL) could introduce a severe vulnerability. The majority of Ethereum nodes are also hosted by a single CSP, namely Amazon Web Services (AWS). Consequently, both the hardware and networking layers are effectively highly dependent on the honesty and availability of the operations of a single organization. Ethereum arguably serves as a role model in pushing for a high diversity of implementations of blockchain node software, e.g., by independent teams building on different underlying programming languages. Still, as of mid-2023, almost 60 % of Ethereum nodes run the Geth execution client (Ether Alpha, 2023), exposing the entire network to bugs in this client or supply chain attacks on underlying Go libraries. On May 11, 2023, the Ethereum blockchain experienced a temporary halt in block finalization due to a shared bug that impacted two consensus clients, collectively responsible for over 33 % of the network's operations (Pereira, 2023). This incident illustrates the potential risks of centralization on the software layer. Finally, economies of scale for PoW (Arnosti and Weinberg, 2022), the general centralized distribution of capital, and the existence of entry barriers to node operation and participation in both PoW and PoS consensus have concentrated mining and staking activities among a small number of entities ("whales" and "pools"), making permissionless blockchains prone to centralization also on the consensus layer. Sai et al. (2021) develop a comprehensive taxonomy that structures these and more centralization aspects of blockchains and describe how they cause increased vulnerabilities, from less harmful denial-of-service attacks and censorship opportunities to threats on the integrity guarantees of blockchain networks that can facilitate even worse incidents like double-spending attacks. Similar developments can also be observed for many decentralized autonomous organizations (DAOs) that govern complex smart contract-based applications: Despite their self-advertisement as decentralized (Barbereau et al., 2023), both voting rights and voting activities are often highly centralized, such that single entities can sometimes easily take complete control. Recently, an incident in which an individual took control of the popular blockchain-based mixing application "Tornado Cash" illustrated these risks (Stradbrooke, 2023). Kannengiesser et al. (2021) and Schlatt et al. (2022a) explore a variety of attack opportunities on blockchains and smart contracts on the infrastructure and application level, many of which are related to centralization or typical pitfalls such as the secure management of cryptographic keys by end users and organizations.

Second, many blockchain-based implementations are still immature and poorly audited, particularly on the smart contract level where bugs, such as vulnerabilities to re-entrancy attacks or a lack of authorization checks, may be present (e.g., Qin et al., 2021; Wan et al., 2021). The integrity guarantees ("safety", "immutability") of blockchains further aggravate these issues, as they make patches only possible under appropriate governance measures defined already a priori (at contract deployment), which introduces a trade-off as it requires a certain degree of trust in the corresponding governing entities, e.g., DAO (Kannengiesser et al., 2021). The combination of determinism and transparency as well as the composability of smart contracts can also introduce additional vulnerabilities on the application layer. In DeFi – "a decentralized financial system that enables financial services and instruments to be offered and used without the need for intermediaries as the system is based on public blockchains and smart contracts" (Gramlich et al., 2023, p. 11) – these risks have manifested in the form of sophisticated attacks, for instance, through combinations of Flashloans (a loan that must be paid back immediately in the same transaction), decentralized exchanges, and lending platforms that integrate low-liquidity price oracles without time-weighted averaging to determine their collateralization requirements and liquidation strategies (Qin et al., 2021). Moreover, determinism and transparency also facilitate value extraction from users to blockchain nodes owing to block proposers' control of transaction ordering, called maximal extractable value (MEV) (Daian et al., 2020).

Even though their degree of decentralization may be lower than anticipated, large permissionless blockchain networks such as Bitcoin and Ethereum have not faced major security incidents on the base layer for a long time. On the other hand, centralized IT systems are not free from substantial technical risks either (considering, for instance, the Log4J vulnerability or the SolarWinds attack), such that these aspects seem relevant yet not critical for designing blockchain-based solutions in organizations. Moreover, the careful design and auditing of smart contracts, both from a technical and game-theoretic perspective (e.g., through an analysis of potential threats from composing them) and appropriate counter-measures can often mitigate these issues (Kannengiesser et al., 2021; McMenamin and Daza, 2023; McMenamin et al., 2022). As such, many of these challenges may be considered less a fundamental characteristic of the underlying blockchain networks but more related to the relative novelty of the technology. Good software engineering practices can arguably address them. Moreover, many application areas for blockchain technology in organizations, like coordinating cross-organizational workflows (Fridgen et al., 2018c), focus on interactions between accountable entities, where some threats can be mitigated also by appropriate risk management on the governance layer (e.g., contractual agreements or insurances), making the security risks less daunting than in pseudonymous cryptocurrency-based payments and DeFi services.

Consequently, it may appear less surprising that an extensive interview study conducted in research paper 5 among organizations that have implemented blockchain-based solutions or that are planning to do so mostly refer to orthogonal technical challenges. One important aspect to consider is that the components in existing blockchain solutions are often very challenging to integrate with legacy systems and cloud-based data pipelines. One reason is that permissionless and permissioned blockchains systems often integrate simple key-value store databases such as LevelDB (e.g., Ethereum) or CouchDB (e.g., Hyperledger Fabric). Therefore, they are not directly compatible with cloud-based storage solutions like Amazon S3, or do not cover particular functionalities demanded by some enterprise applications, such as the graph-based databases often used in modeling supply chains (Leveling et al., 2014). Enterprises hence need to hand-craft their own connectors and data pipelines for the integration of blockchain-based solutions to make sure that the storage of a node does not reach its limit during operation and that the ledger and/or state of the node is continuously copied to databases that are more suitable for the queries of their applications. Yet, while corresponding implementations may be costly, particularly for organizations whose core products and services are not centered in IT, these challenges are relatively straightforward to address conceptually. Consequently, like in the case of the aforementioned security aspects, I do not consider integration a *fundamental* technical barrier either.

This perspective leaves the three main areas of challenges I presented in Section I for which it is less clear how to address them: (1) high electricity demand, (2) low performance accompanied by high operating costs, and (3) excessive transparency and a lack of access control. The following Sections III.2 to III.4 will describe each of these challenges in detail and assess their individual severity. I will argue that while high electricity consumption can be considered an issue that only affects a very specific family of blockchain implementations, performance and data visibility issues are more fundamental in nature as they are directly related to the technical core characteristic of blockchain networks, namely replicated information processing. I will also discuss why existing permissioned blockchains can alleviate some of these issues but still fall short of thoroughly meeting organizations' performance and access control requirements.

III.2 Electricity consumption

The electricity consumption of cryptocurrencies and blockchain-based information systems in general is a frequently discussed topic not only in businesses and public institutions but also in research. There are now more than 50 academic publications that investigate the energy consumption of one or several selected blockchains or that survey the energy consumption of blockchain technology in general (Lei et al., 2021; Sai et al., 2021) or sustainability-related aspects of their operation, such as e-waste (de Vries and Stoll, 2021). Early work focused on the electricity consumption and carbon emissions caused by Bitcoin (e.g., de Vries, 2018; O'Dwyer and Malone, 2014; Stoll et al., 2019; Truby, 2018)), Bitcoin and Ethereum (Cambridge Centre for Alternative Finance, 2020a; Cambridge Centre for Alternative Finance, 2020b; Digiconomist, 2022), or a select number of PoW cryptocurrencies (research paper 1, Gallersdörfer et al., 2020; Krause and Tolaymat, 2018). Research paper 1 was the first to consider the energy consumption characteristics of blockchain technology in general, including a perspective on alternative consensus mechanisms for permissionless blockchains and permissioned blockchains.

PoW blockchains are energy intensive by design, with their electricity consumption reaching the level of smaller industrialized countries (research paper 1; Gallersdörfer et al., 2020). The key reason is the significant rise in economic incentives to engage in mining activities following increasing cryptocurrency prices: In most PoW blockchains, the proposer of a new block is rewarded with a certain number of newly generated cryptocurrency coins and with additional transaction fees that are naturally required to balance the demand for transaction processing with limited supply for block space (Ilk et al., 2021; Roughgarden, 2021). As in PoW, the probability of proposing a block is proportional to a miner's share of computational power invested (Arnosti and Weinberg, 2022; Budish, 2018), these developments have incentivized significant investments in mining hardware and electricity to participate in the race for finding new blocks and earning corresponding rewards.

Although the electricity consumption of PoW blockchain networks cannot be observed directly, a combination of technical and economic considerations allows obtaining lower and upper bounds or best-guess estimates that localize the electricity consumption reasonably well. First, both a lower bound and a best guess on electricity consumption can be obtained through a *technical* approach (e.g., de Vries, 2018; Gallersdörfer et al., 2020; O'Dwyer and Malone, 2014; Vranken, 2017): From observing the complexity of the cryptographic puzzles solved in PoW and the number of found solutions (i.e., the frequency at which new blocks are added), one can estimate the number of solution attempts.[7] A lower bound on the amount of electricity needed can be determined by identifying the most efficient mining hardware on the market. Likewise, obtaining an empirical estimate of the precise mining hardware distribution yields a best guess for electricity consumption. On the other hand, taking an *economic* perspective allows determining an upper bound or a best guess for electricity consumption, as rational miners will only operate when their expected revenues exceed their expected operational costs (see, e.g., Budish, 2018; de Vries, 2018; Hayes, 2015; Vranken, 2017). By estimating a lower bound on global electricity prices or a corresponding distribution, as well as the share of operational costs for hardware procurement and infrastructure maintenance, an upper bound and a best guess for electricity consumption can be obtained. Some publications, such as Gallersdörfer et al. (2020), Krause and Tolaymat (2018), and Stoll et al. (2019), also applied hybrid approaches, for instance, by first removing older, less efficient hardware that is no more profitable from the distribution obtained by sales data for mining hardware

[7] Under the plausible assumption that brute-force is the most efficient approach and that miners engage in these brute force attempts with independent randomness ("nonces").

(this part corresponds to applying the economic approach for determining an upper bound on the micro-level) and then using the refined mining hardware distribution for obtaining more accurate results when applying the technical approach.

Applying these lower- and upper-bound estimates to a variety of PoW-based cryptocurrencies beyond Bitcoin yields two key insights: First, even in a large network with tens of thousands of nodes, the energy consumption caused by PoW consensus is several orders of magnitude larger than the electricity consumption caused by the replicated information processing, i.e., operating blockchain nodes. Second, the dominance of Bitcoin in terms of market capitalization is mirrored by its dominance in terms of energy consumption (research paper 1; Gallersdörfer et al., 2020). All other PoW cryptocurrencies account for less than 50 % of the electricity consumption of Bitcoin. Moreover, for cryptocurrencies like Bitcoin that are subject to "halving events" or other reductions of the number of new coins rewarded to miners over time, a stabilization of the corresponding cryptocurrency prices and transaction fees implies a long-term decrease in electricity consumption, given non-decreasing electricity prices (research paper 1). Yet, as the economic security of PoW blockchains is determined by the costs of electricity and hardware used for mining operations, it is contestable whether long-term reductions are to be expected: It is conceivable that developers will intervene with protocol changes to increase the security budget, which inevitably increases energy consumption.

As such, and arguably counter-intuitively, the energy consumption of PoW blockchains is not driven by technical inefficiencies or complex cryptographic operations in the first place, but instead by a deliberate design choice that provides security through the economic value of electricity and hardware used for mining operations. This perspective reveals that assuming a linear relationship between the number or complexity of processed transactions and total electricity consumption, as suggested, for instance, by Mora et al. (2018), is a misconception that – because a linear relationship seems a reasonable assumption for IT systems – is unfortunately widely believed. The economic approach also suggests that in the long run, i.e., in economic equilibrium, the energy efficiency of mining hardware does not have an impact on the energy consumption of PoW blockchains.

Consequently, while PoW blockchains consume enormous amounts of electricity, there is little reason to expect that electricity consumption will dramatically change in either direction in the future. Considering the highly limited transaction processing capabilities of existing permissionless blockchains (see Section III.3.1), and also corresponding carbon emissions (Stoll et al., 2019) and e-waste (de Vries and Stoll, 2021) caused by mining operations for PoW blockchains, it thus seems that the criticism of the sustainability of

Bitcoin and other PoW blockchains is legitimate (Goodkind et al., 2020; Truby et al., 2022).

On the other hand, the energy consumption of PoS blockchains and permissioned blockchains is not driven by economic relations, as they do not involve any computationally heavy activities for their functionality and security (research paper 2; Gallersdörfer et al., 2022). As consensus-related operations are typically relatively simple, such as a certain number of highly efficient digital signature verifications (Saleh, 2021), their electricity consumption is caused mainly by the replicated processing of transactions. In fact, running a node in PoS blockchains that aim at a high degree of decentralization, like Ethereum, requires only little computational power, which can be provided by hardware like a Raspberry Pi with low electric power consumption. Consequently, the electricity consumption of these networks is more than three orders of magnitude lower than that of Bitcoin and in particular than that of other PoW blockchains (research paper 1; Gallersdörfer et al., 2022).

The energy consumption of permissioned blockchains can be considered even more moderate, as these systems tend to involve a substantially smaller number of nodes than permissionless blockchains. The potentially higher energy consumption of these nodes to meet higher processing requirements is balanced by the also higher number of transactions that can be processed (Platt et al., 2021b). This perspective is supported through measurements of the performance of different centralized databases and permissioned blockchains with the DLPS (research paper 2), which suggest that cryptographic and communication overhead in a blockchain network makes them around two orders of magnitude less energy efficient than centralized databases, but still significantly more energy efficient than permissionless PoS blockchains (and trivially several orders of magnitude more energy efficient than permissionless PoW) blockchains.

Figure 2 summarizes this discussion by presenting a ballpark estimate for the energy consumption of different blockchain networks. It opts to display electricity consumption per transaction, which is a controversial and sometimes misleading metric for PoW-based blockchains (research paper 1; Carter, 2021) but more appropriate for the comparison of centralized databases and non-PoW-based blockchains when idle consumption is less significant than total electricity consumption. These ballpark estimates were initially given by research paper 1 and have later been supported by measurements (research paper 2; Gallersdörfer et al., 2022; Gallersdörfer et al., 2020; Platt et al., 2021b). Consequently, for non-PoW blockchains, energy consumption is moderate enough that one can expect net savings in many applications in which a blockchain-based information sys-

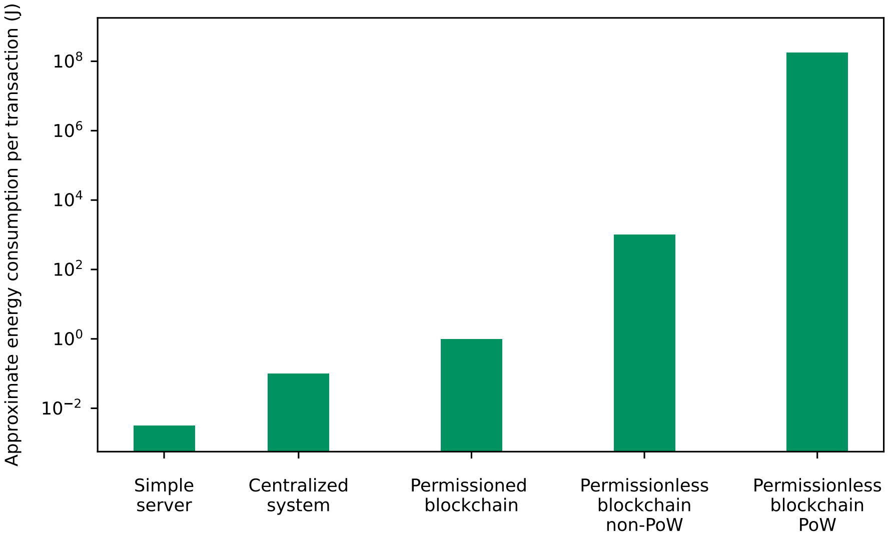


Figure 2: Ballpark comparison of the electricity consumption of centralized systems and different blockchain networks (adapted from research paper 1).

tem facilitates the improvement or digitalization of business processes (see, e.g., research paper 2). I conclude that when avoiding PoW blockchains, electricity consumption and sustainability in general seem hardly problematic already for existing blockchain-based solutions.

While the energy consumption of PoW blockchains is significant, there is neither an unavoidable trajectory toward higher nor toward lower energy consumption in the future. On the other hand, the economic nature of the high energy consumption of PoW blockchains and the design choice to anchor the security of PoW blockchains to investments in the hardware and electricity required for mining operations leaves no hope that their energy consumption can be substantially reduced through technological improvements. Consequently, neither doomsday scenarios nor inactivity seem appropriate. Related research and practitioners controversially discuss whether moving to alternative blockchain designs in permissionless blockchains that do not build on the energy-intensive-by-design PoW is the right approach. While the remaining degree of redundancy in these non-PoW blockchains still introduces some inefficiencies, these inefficiencies can easily be compensated by the replacement of non-digital workflows or efficiency gains in cross-organizational processes (as, for example, discussed in research paper 2). Moreover, in contrast to PoW consensus, the remaining inefficiencies in PoS-based permissionless blockchains or permissioned blockchains are purely technically caused. Consequently, improvements both on the blockchain layer (e.g., state channels, optimistic rollups, validity rollups, and sharding and on the hardware layer (improved energy efficiency of CPUs and networking components) will automatically translate into further reductions in energy consumption. As such, when avoiding PoW-based blockchain solutions, energy consumption issues of blockchain technology seem to remain an issue that is better considered in organizations' public relations departments rather than the IT department.

Yet, PoW is often considered more secure than alternative consensus mechanisms for permissionless blockchains (e.g., Zheng et al., 2017). A common argument that PoW may be more secure is that its implementation seems less complex than implementing PoS. For instance, the development and implementation of the PoS consensus mechanism of Ethereum took several years to handle all known attack scenarios stemming from the "costless simulation" – a consequence of the avoidance of excessive electricity needs (Schwarz-Schilling et al., 2022). On the other hand, there are also good arguments why the costs to deploy a successful attack on PoS systems could be much higher than the costs to attack even the largest PoW-based blockchain, Bitcoin: The security of PoW blockchains is tied to the economic value of newly created tokens (coins) and current transaction fees, whereas the security of PoS blockchains is tied to the economic value of the whole network and transaction fees (Buterin, 2020). There are also indications that PoW blockchains may be subject to stronger centralization tendencies that may make attacks on its integrity more practical in the long term compared to PoS (Arnosti and Weinberg, 2022; Roşu and Saleh, 2021). Consequently, the statement that PoW is fundamentally more secure than PoW seems at least debatable. On the other hand, permissioned blockchains generally base their security on the assumption that no more than a certain fraction of the number of nodes (often $\frac{1}{3}$) will collude, behave maliciously in another way, or be successfully attacked by an adversary. Under this assumption, the integrity guarantees are essentially connected to the security of cryptographic primitives like hashing algorithms and digital signatures that underly every digital interaction that we do on the web today. Industry consortia that trust a certain share of a consortium or that agree on contractual penalties may even feel safer in such a permissioned setting than relying on a broad distribution of mining power (PoW) or capital (PoS) in a large permissionless networks, which is generally not fully transparent. This observation may be one of the key reasons that many public- and private-sector projects and research proposals have opted for building on non-PoW blockchains: They prioritize sustainability aspects over unclear gains in integrity guarantees.

Several studies indicate that the low energy consumption of these non-PoW-based blockchain solutions implies that the energy savings potential of blockchain-based applications can easily exceed the remaining inefficiencies that these systems exhibit through their replicated processing of transactions (research paper 2; Köhler and Pizzol, 2020;

Sislian and Jaegler, 2022). As such, the supposedly high electricity consumption of blockchains can be considered fully addressed by blockchain solutions that are sufficiently secure and mature and, therefore, available for adoption by organizations. Moreover, improvements in hardware efficiency as well as scaling solutions (see also Section III.3.2), will further improve the energy efficiency of these non-PoW blockchain designs in the future. Hence, I conclude that energy consumption aspects should not be considered a fundamental challenge of blockchain adoption by organizations.

III.3 Performance

III.3.1 Problematization

Organizations often have considerable requirements for the performance of their IT systems. Considering performance aspects can involve many dimensions. A generally established metric to assess the performance of an IT system is *maximum throughput* (Jain, 1991) – the number of transactions that a blockchain can perform in a given time frame, typically measured in transactions per second. Organizations' throughput requirements vary considerably. For instance, the branch of a single bank arguably only needs to handle a few requests per second or even per hour in a small municipality. Yet, blockchains are considered most promising in scenarios in which they unite many stakeholders at large scale (Fridgen et al., 2018c; Jensen et al., 2019), and should therefore be mapped to the requirements of IT systems such as international trade systems, international payment systems, or large e-commerce platforms. For instance, the TradeLens platform has processed hundreds of millions of container shipments when it was still operational, with each container comprising potentially dozens of shipping events involving multiple stakeholders such as logistics service providers, customs, ports, and banks (Fridgen et al., 2019). Even at the still moderate scale of the TradeLens platform, this corresponded to at least a mid-double-digit number of transactions per second required. To give another example, the VISA network operates around 2 000 payments per second on average and up to 25 000 payments per second in peak times globally (VISA, 2019). Yet, this is still a small figure compared to the peak throughput requirements of even larger-scale systems. For example, on Amazon Prime Day, Amazon's core database needed to handle up to 45 million requests per second (Barr, 2019). Throughput requirements may also depend on the complexity of the transaction logic under consideration, the size of data that needs to be up- or downloaded, etc. For example, several hundreds of gigabytes of data are uploaded to YouTube every minute, which corresponds to an upload rate of several gigabytes per second.

Today, centralized applications with high throughput requirements can be run relatively easily by procuring appropriate hardware or cloud instances and using the load balancing capabilities provided by container orchestration tools like Kubernetes (Bernstein, 2014). Notably, this approach can still cause substantial challenges regarding cost structure: When a business procures expensive and electricity-demanding hardware that runs at low utilization close to idle state most of the time just to be prepared to scale to meet peak demands at specific times (451 Research, 2019), they pay for the expensive hardware all the time. *Elastic scaling* with an associated pricing structure that linearly depends on the capacity in use is hence often desirable for organizations that aim to scale their operations rapidly (research paper 5). Hyper-scale CSPs can use the law of large numbers when distributing their computational capacity dynamically to their business users' load balancing and so reduce their ratio of peak to average utilization (Shehabi et al., 2016). Ultimately, this allows business users to pay for used capacity instead of reserved peak capacity. Serverless computing provides an alternative but similarly scoped approach that can shield even more scaling-related complexities from developers and provide further improved cost structures as there is a purely linear relationship between computational resource requirements and costs. More precisely, costs for performing certain operations are proportional to their number as well as the corresponding execution time and memory requirements (AWS, 2021). In serverless computing, the task of providing enough computing power, bandwidth, and storage for this elastic scaling is completely outsourced, making computation available as a fully managed service provided by hyper-scale CSPs.

Another crucial key metric to consider in organizational information systems is latency, which determines how long it takes on average after submitting a client request until the corresponding transaction can be considered irreversibly included in the state of the system (research paper 1). Centralized IT systems often provide latencies between few milliseconds and a few hundred milliseconds; with additional network delays emerging between distant points on Earth that can also be on the order of a few hundred milliseconds. Consequently, end users and business users are used to expecting also a high level of responsiveness and real-time capabilities of their IT systems (research paper 5).

A common perspective (which one could even call a misconception) on the performance of blockchain-based systems is that it is the high computational effort required for PoW that makes the performance of PoW blockchains very low. Bitcoin, for example, can handle only around 7 transactions per second and has a latency of around one hour when substantial finality guarantees are required (Georgiadis, 2019; Nakamoto, 2008). However, at second glance, it becomes apparent that the PoW consensus is not responsible for the low throughput of Bitcoin in the first place: At the latest since CPU mining has become non-competitive, the computationally intensive mining process is entirely outsourced to dedicated mining hardware, with application-specific integrated circuits (ASICs) arguably being the most prominent representatives (de Vries, 2018). Another argument that supports the view that mining is not the core reason for the poor performance of PoW blockchains is that when increasing block size by a small factor, the mining process would not change considerably, and yet maximum throughput would multiply. Only when taking a closer look, the design choice of PoW does indeed have some minor implications on performance: To use the resources of a node (i.e., CPU, memory, disk i/o, storage, and bandwidth) most effectively, a very short block time would be optimal. However, block time cannot be reduced arbitrarily in PoW without compromising security because a short block time implies an increased occurrence of orphan blocks that reduce the security guarantees of the network (Gervais et al., 2016).

In general, the main reason for the low throughput of permissionless blockchains and the efforts made to increase performance is largely independent of the consensus mechanism: To maintain a high level of decentralization and, thus, security (see Section III.1), permissionless blockchains aim to make the barriers to participation by running a node as low as possible (Buterin, 2021). While permissionless blockchains do not include any form of registration process that would pose formal barriers to this kind of participation, an implicit prerequisite to participation for an entity is access to the hardware (CPU, memory, disk, storage) and bandwidth resources necessary to maintain a synchronize a node. For instance, even with a peak throughput of only around 7 transactions per second on Bitcoin, after more than 13 years of operation, a Bitcoin node must provide almost 500 GB of storage as of mid-2023. On the other hand, participation in the high-performance “permissionless” Solana blockchain demands hardware with at least 12 CPU cores at 2.8 GHz, at least 128 GB RAM, and a bandwidth of at least 300 MBit bot for up- and download (Solana, 2023) to reach an alleged throughput of 65 000 transactions per second. Assuming consistent maximum utilization and that this throughput consumes approximately the entire bandwidth required, each of the around 3 000 participating nodes in the Solana network would need to supply more than 25 TB of storage *every day*. In more general, compute, memory, disk, and storage requirements are directly tied to local costs of deployment; moreover, on-premise storage limitations may require developing sophisticated data pipelines for high storage requirements. Bandwidth requirements, on the other hand, are not only related to costs but also tied to the availability of potentially complex networking infrastructure, such as fiber-optic cables. These requirements thus impose high entry barriers and, ultimately, lead to centralization in terms of technical diversity and geographical and social distribution, with the corresponding negative security implications (Section III.1; Sai et al., 2021). This is why the definition of “scalability” usually refers to throughput when given a fixed hardware setup or budget (e.g., Starknet, 2021). In essence, the performance limitations of permissionless blockchains are a deliberate design choice that demands low resource requirements for participation as a node to achieve high degrees of decentralization and, ultimately, security. Besides demanding high computational requirements, there are also many other scaling solutions that were discussed relatively early, such as execution sharding (Zamani et al., 2018) and application-specific blockchains that are interconnected through “bridges” (Lee et al., 2022). Execution sharding relies on the contested assumption that only a minor share of transactions involves multiple shards at the same time and at the time mostly remains a theoretical concept. Moreover, bridges typically trade transaction processing capacity improvements against additional attack vectors and a lower degree of decentralization, as they often rely on a single trusted notary or a small number thereof. Bridges that implement algorithmic verification of PoW blockchain consensus (Bünz et al., 2020) or PoW consensus (Gaži et al., 2019) on the two connected blockchain have only recently become practical.

When the limited throughput of permissionless blockchains meets high demand for transaction processing capacities, transaction fees become substantial (Ilk et al., 2021). Even simple transfers on the Ethereum blockchain have cost a double-digit euro amount for most of the time in the last few years. For more complex operations, such as deploying smart contracts for business processes and interacting with them, costs can quickly become one or two orders of magnitude higher. This situation makes popular permissionless blockchains hardly suitable for meeting the cost expectations of transaction-based enterprise information systems, which thus far have operated on centralized databases where processing a simple transaction can cost as little as 10^{-9} €.[8]

For permissioned blockchains that underly many cross-organizational networks, the degree of decentralization plays a smaller role in design decisions. In these settings, stakeholders are also more likely to afford substantially higher hardware and networking in-

[8] According to own experiments, a simple database like LevelDB can manage several thousands of transactions per second on a server with at most 50 W of power consumption. Assuming electricity costs of 0.3 $\frac{€}{\text{kWh}}$, this corresponds to the given figure.

frastructure quality. Yet, the consensus mechanism and sequential operations involved in node operation suggest that beyond storage and bandwidth, also computation and disk resources may limit the throughput these solutions can achieve. Consequently, there has been a lot of research on the performance of common permissioned enterprise blockchain solutions, such as Hyperledger Sawtooth (e.g., Perboli et al., 2020; Shi et al., 2019), Quorum (e.g., Baliga et al., 2018a; Mazzoni et al., 2022), and Hyperledger Fabric (e.g., Androulaki et al., 2018; Thakkar et al., 2018; Wang and Chu, 2020). Yet, there is little related work that determines the maximum performance of permissioned blockchains in real-world rather than in toy scenarios, e.g., by conducting sensitivity analyses for all blockchain network parameters that may be modified in an implementation in organizations. Nevertheless, there are some notable examples. Baliga et al. (2018b) and Thakkar et al. (2018) study the impact of the underlying database performance on the overall blockchain layer. Moreover, the availability of many different benchmarking frameworks with a lack of documentation on how the key performance metrics are defined and measured makes these performance evaluations difficult to compare. The DLPS developed in research paper 3 provides a benchmarking framework that supports flexible comparisons of the performance of different permissioned blockchains with well-defined metrics. It develops formal definitions of maximum throughput and latency and implements an algorithm that retrieves these metrics automatically after executing startup scripts that bootstrap different permissioned blockchains in configurable setups (e.g., number of nodes, choice of node hardware, transaction payload). Using the DLPS, a comprehensive benchmarking study of multiple permissioned blockchains that are commonly used in enterprise proofs of concepts, pilots, and production implementations reveals that the maximum throughput that established permissioned blockchains can achieve is indeed one to two orders of magnitude higher than for the popular permissionless blockchains, reaching a throughput of up to several thousand transactions per second (see Figure 3). However, maximum throughput is still significantly lower than in centralized systems supporting the said millions of transactions per second (Barr, 2019). On the other hand, Figure 4 illustrates that the permissioned blockchains investigated exhibit a relatively stable latency of around 2 seconds, which may not be sufficient for all applications in which information must be included on a blockchain in real-time but still seems practical in many scenarios.

A case study of Hyperledger Fabric also demonstrates that the choices of network size (number of nodes) and network topology (which determines, for instance, network latencies caused by intercontinental deployments) have only a moderate impact on transaction latency and throughput (research paper 4; Androulaki et al., 2018). Yet, computation-

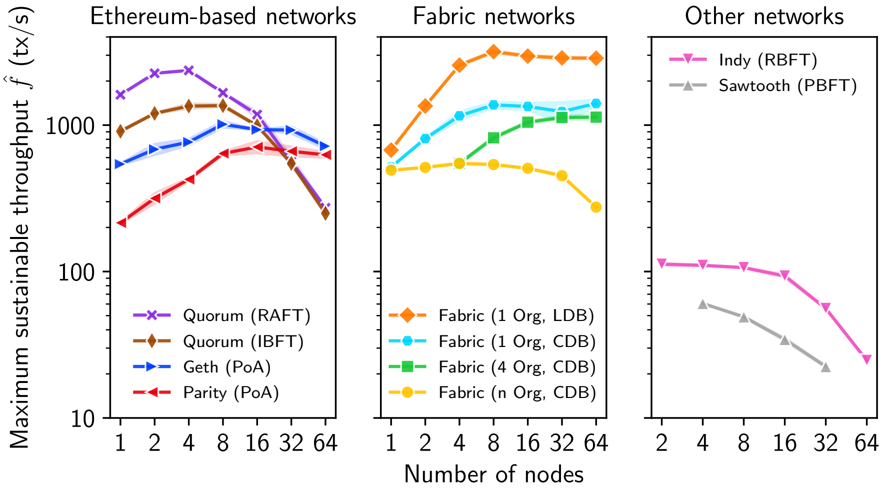


Figure 3: Comparison of maximum throughput for selected permissioned enterprise blockchains and different network sizes (Figure taken from research paper 3).

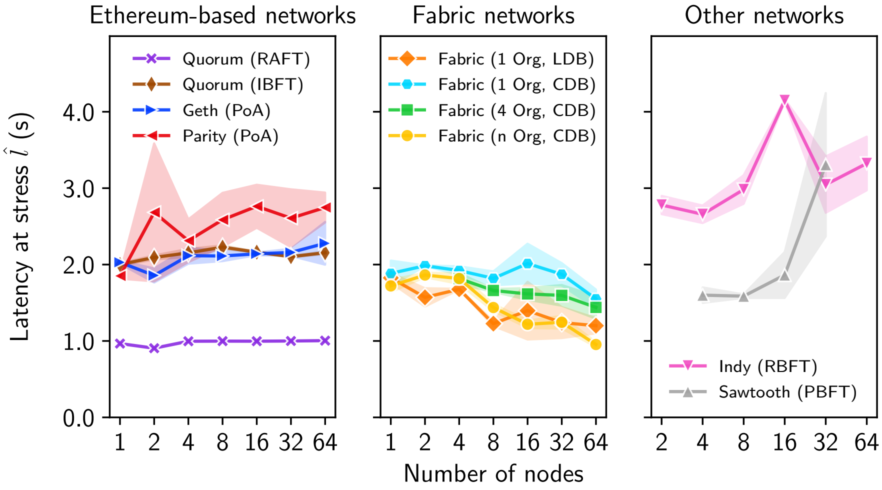


Figure 4: Comparison of latencies for selected permissioned enterprise blockchains at 80 % of maximum throughput for different network sizes.

ally intensive and data-intensive workloads or the enhanced privacy features Hyperledger Fabric offers (see also Section III.4.1) can reduce maximum throughput substantially. For example, measurements by research paper 4 suggest that when transactions involve the processing of only 100 kB of data, throughput can degrade to only around 20 transactions per second; moreover, the maximum data throughput is limited to around 10 MB per second. The experiments also illustrate that these performance degradations can only be addressed to a minor extent through more powerful and, thus, expensive hardware because of the limited opportunities for multi-threading in some of the blockchain-specific sequential workloads. Particularly for computationally and data-intensive tasks like cross-organizational data lakes, performance and cost issues hence seem substantial (research paper 7).

In sum, the intentionally lower and controlled degree of decentralization and replication through high hardware and bandwidth requirements allow permissioned blockchains to achieve substantially better performance than their permissionless counterparts. However, as permissioned blockchains are still characterized by replicated processing, they are hardly suitable for high-throughput scenarios, particularly when supporting computationally and data-intensive applications, or for providing extremely low latency.

III.3.2 Solution approaches

The performance limitations of both permissionless and permissioned blockchains described in Section III.3.1 have attracted significant attention from researchers and practitioners. In the realm of permissionless blockchains, the highly limited supply of block space on the one side and the substantial demand for transaction processing on the other side have thus led to what is known as the "transaction fee crisis" (Ilk et al., 2021). Permissionless blockchains have explored increasingly complex approaches toward addressing this challenge. As these blockchain networks generally aim to further increase their adoption, increasing the supply with transaction space seems the more popular approach. Related research has also developed some ways to incrementally improve the performance of permissioned blockchains. For example, the throughput of Hyperledger Fabric can be further increased by some optimizations that improve the degree of parallel resource utilization, i.e., multi-threading (Thakkar and Natarajan, 2021) and via keeping the state in RAM (Gorenflo et al., 2020). Yet, these configurations come with some tradeoffs (e.g., considerable RAM requirements and a loss of the state when a node crashes). Moreover, when aiming for a maximum throughput of tens of thousands of transactions per second, bandwidth requirements as well as storage requirements in the long run become prohibitive (research paper 4). Also for permissionless blockchains, simple solutions such as "pruning" that allow deleting old transactions from the ledger have been discussed, but have only rarely been applied because of the need to provide nodes that join a permissionless blockchain with a complete history for objective verifiability.

As discussed in Section III.3.1, the "holy grail" of increasing the performance of permissionless blockchains and, thus, reducing high operating costs and transaction fees, is to increase the transaction processing capacity of the blockchain network while avoiding (1) higher resource requirements for every node and (2) additional components that involve new trusted stakeholders, system discontinuities and, ultimately, novel vulnerabilities to a maximum extent. Intuitively, this seems a daunting exercise, as the replicated processing that characterizes blockchains seems to demand that every node processes the transaction by every participant in the network. This situation is also known as the "verifier's dilemma" (Luu et al., 2015). It turns out that there are still several approaches to scale permissionless blockchains that only involve very moderate additional trust assumptions. These solutions are typically called "Layer-2 solutions" (Gudgeon et al., 2020). Prominent examples include relatively simple constructions, including payment channels, payment hubs, and routing solutions such as the Bitcoin Lightning Network. These solutions facilitate repeated transfers between the same entities without requiring an on-chain transaction for each of these transfers (Poon and Dryja, 2016). However, these solutions are only feasible for very restricted process logic, such as a simple transfer. For general-purpose transactions as provided smart contracts, other solutions are required. Another approach is based on trusted execution environments (TEEs). TEEs describe a particular area in a CPU that can securely store cryptographic keys that cannot be retrieved even by an entity with physical access to the corresponding chip. Similarly, TEEs can protect computations executed inside them against corruption attempts. Using the keys they protect, they sign the results of computations inside them to provide attestations of the correctness of code executed in them. Consequently, blockchain nodes can rely on attestations provided by another node's TEEs that certain (batches of) state transitions were applied correctly without doing the computation on their own, which improves performance significantly (Fang et al., 2022). Unfortunately, in practice, TEEs require trust in corresponding CPU manufacturers. Moreover, they have proven vulnerable to attacks (Muñoz et al., 2023) and corresponding project discontinuation (Pezzone, 2022). The initial setup processes of the TEE which involve key generation and distribution can also be complex, specifically when TEEs of different manufacturers must be integrated and when large states that do not fit in the limited memory of a TEE need to be managed. As a consequence, researchers and engineers mainly consider approaches that can provide attestations of correct execution without relying on TEEs.

Arguably the most popular and promising of these solutions are optimistic and validity rollups (Thibault et al., 2022). In optimistic rollups, computations can be outsourced to third parties that record the result on-chain and provide a collateral to be held accountable for the right computation. Within a certain dispute period, any participant in the blockchain network (typically a node) can challenge this result and claim a part of the collateral if they detect fraud (Adler and Quintyne-Collins, 2019). This approach introduces only minor additional security assumptions, namely the presence of a single honest participant that cannot be censored for the duration of the dispute period. However, it faces some issues regarding incentive-compatibility (a lack of incentive for third-party verification efforts as long as the third party is honest) and efficiency (long time to finality, which can involve liquidity issues) (Gluchowski, 2019). Moreover, the throughput improvements are still limited as all input and output data corresponding to transactions, including smart contract invocations, still need to be recorded by all nodes. Yet, the latter issue can be addressed to some extent with sharding data storage, as the presence of a single honest node in a shard is sufficient to guarantee data availability – in contrast to computational integrity that requires a majority. Validity rollups aim to address some of the remaining challenges of optimistic rollups. While they build on substantially more complex concepts, in particular succinct cryptographic proofs such as SNARKs (Ben-Sasson et al., 2013)[9], they maintain the verification of every single transaction by every node by default, avoiding incentive incompatibilities, potential censorship threats, and long time to finality (Gluchowski, 2019). The cryptographic validity proofs also allow removing substantial parts of data (such as the digital signatures associated with individual transactions) and, therefore, higher degrees of compression (Lavaur et al., 2023). Yet, the high cryptographic complexity of validity rollups also introduces new security risks.

Validity rollups for simple transfers have been operational since 2020 already. Prominent examples include Loopring[10] and Hermez[11] and can increase the throughput of a permissionless blockchain by at least two orders of magnitude, which also implies a reduction of transaction fees by roughly the same factor. On the other hand, general-purpose implementations that can cover arbitrary smart contract interactions have only emerged in 2023, such as the Polygon zkEVM[12] and Scroll[13]. Consequently, to date, relatively mature solutions exist for simple transfers and situations that are much more

[9] Originally, SNARKs were used for the first time in cryptocurrencies in the form of zero-knowledge SNARKs (zk-SNARKs) for improving privacy, most importantly, in Zcash (Ben-Sasson et al., 2014). Although succinct cryptographic proofs and in particular SNARKs can often be made zero-knowledge with little additional effort, many SNARKs used in scalability solutions are in fact not zero-knowledge, for instance, the ones used in Starkware's validity rollup (https://starkware.co/starkex/). The core property needed for compressing computation and, therefore, improving scalability is their succinctness. On the other hand, many popular frameworks for implementing SNARKs, such as implementations of Groth16 (Groth, 2016) in snarkjs (https://github.com/iden3/snarkjs) and arkworks (https://github.com/arkworks-rs/groth16), are also zero-knowledge. Consequently, the terms "zk-rollups" or ZKP are very commonly used by software engineers and researchers as a synonym for SNARKs independent of whether they actually have the "zk" property.

[10] See https://loopring.org.

[11] See https://docs.hermez.io/Hermez_1.0/about/scalability/.

[12] See https://polygon.technology/polygon-zkevm.

[13] See https://scroll.io/.

computationally-heavy than data-heavy and that can tolerate additional long finality, but the more sophisticated solutions that promise substantially higher performance gains, particularly in combination with novel data availability concepts (Yu et al., 2020), are still in their early stage. Notably, as briefly discussed in research paper 1 and also elaborated on in Sedlmeir et al. (2020a), these concepts can reduce the aggregate computational load in the system and, therefore, also further reduce the share of energy consumption of blockchains that is technically caused, which is the dominant contribution in non-PoW permissionless blockchains.

Despite these promising technical developments, the complexity of integrating these scaling solutions in organizations is still significant. Solutions such as optimistic and validity rollups that can integrate with existing smart contract programming languages arguably simplify the adoption of these concepts in organizations, but there seems to be a considerable knowledge gap regarding the different tradeoffs of these solutions and their affordances; particularly in organizations where blockchain alone already was considered highly complex. Consequently, future research will be required to identify how organizations can manage to adopt these scaling solutions without facing further increases in complexity. Re-assessments of previous studies on tradeoffs in blockchain systems, such as Kannengießer et al. (2020), may also be a promising avenue for future information systems research.

On the other hand, borrowing ideas from scaling concepts for permissionless blockchains only has limited potential for enterprise blockchains: First, concepts like state channels and the Lightning Network that allow conducting most interactions in a bilateral way and only do on-chain settlement from time to time can only be applied to highly specific interactions like simple payments, and there is no obvious way to design similar solutions for processes in organizations that involve general smart contract functionality. Second, reducing the degree of replication through approaches like sharding (Zamani et al., 2018) has a much smaller effect when the hardware and bandwidth quality provided in a permissioned blockchain is already high and the degree of replication is not in the four- or five-digit range but in the two-digit range, where a further decrease of the degree of replication may harm availability and integrity guarantees much more than in large permissionless networks. Moreover, the scaling opportunities using rollups are also questionable. First, optimistic rollups rely on the availability of a sufficient number of honest "watchers". Second, the significant increase of latency for finalizing transactions is a tradeoff that seems difficult to harmonize with applications with close to real-time latency requirements (see also research paper 5). On the other hand, approaches based on succinct cryptographic proofs of computational integrity also seem hardly beneficial because they still involve a prover that faces substantial overhead. As a rule of thumb, this overhead can be millions of times more computationally intensive than the original computation (Thaler, 2022) while only a small number of nodes has a significantly smaller computational effort. More formally, if the computational effort associated with a transaction was X in the unmodified blockchain system with a full degree of replication and N the number of full nodes, the computational effort of the whole system is $N \cdot X$. In a validity rollup-based design the effort would be $\eta \cdot X + \varepsilon \cdot N$, where η is the multiplicative prover overhead and ε is relatively small and (in many validity rollups) independent of X. The ratio between total computational effort in the blockchain system with and without a validity rollup is, therefore,

$$f = \frac{\eta \cdot X + \varepsilon \cdot N}{N \cdot X} \approx \frac{\eta \cdot X}{N \cdot X} = \frac{\eta}{N}.$$

Consequently, for permissionless blockchains with N large enough, there can be significant net savings in computational effort for the system. However, for N small, as in common permissioned blockchains, the total computational effort is substantially increased, questioning the fitness of a validity rollup for scaling the network.

Consequently, alternative approaches facilitate elastic scaling through constructions of nodes based on “serverless” components (research paper 5), and introducing additional trust assumptions in a TEE and its manufacturer (Liu et al., 2022) or a small set of CSP (research paper 5). Particularly for scenarios with exceptionally high throughput requirements or where elastic scaling is important (e.g., to keep operating costs low for a startup without the need to re-deploy the blockchain network when scaling the system), a serverless design can be appropriate. This approach constructs all functionalities required from a blockchain node through “serverless” components, i.e., computational or database operations that come as fully managed and, thus, scalable cloud services that are billed per invocation. According to research paper 5, such serverless designs can handle tens of thousands of requests per second, outperforming other permissioned blockchain designs. Moreover, the corresponding cost structure is appealing, as costs are proportional to throughput and can additionally be controlled through a choice of batch size: Operating large batches (“blocks”) reduces the consensus-related costs per transaction substantially, as indicated in Figure 5: For a batch of one transaction, consensus overhead dominates the costs (“Consensus Coreography” and “Verify & Vote”; more than 62.4 %). On the other hand, For the maximum batch size of 900 (this is the limit in the implementation considered in research paper 5), the major share of a transaction indeed relates directly to the

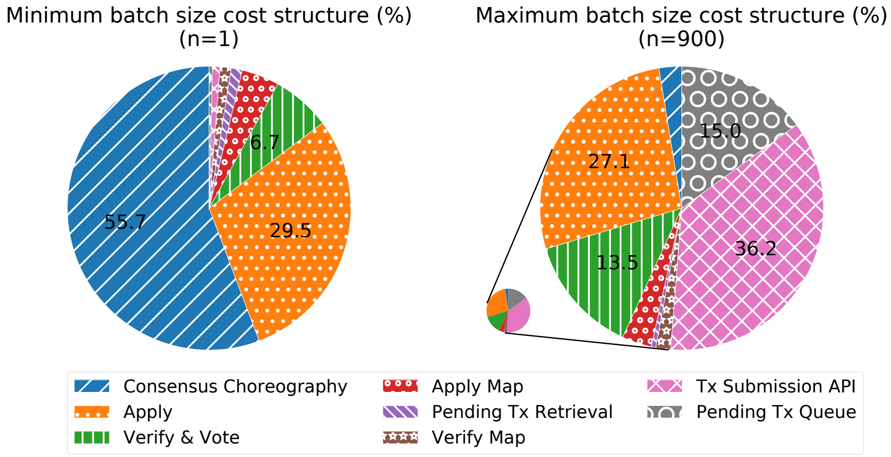


Figure 5: Cost structure of serverless blockchains depending on the batch size.

submission of the transaction ("Tx Submission API" and "Pending Tx Queue"; 51.2 %) – costs that can hardly be avoided or reduced. However, the substantial benefits in terms of performance and operation costs come with the tradeoff that these serverless designs rely on a CSP that provides the corresponding serverless components and that is trusted with regard to correct execution and high uptime. Yet, as research paper 5 discusses, this tradeoff may be acceptable for many companies that already have a contractual agreement with a CSP, such that the major limitation may be the small number of CSP that offer the necessary components and, thus, the high degree of centralization.

In sum, approaches to improving the performance and in particular the throughput of blockchains while keeping operation costs bounded are manifold and follow very different trajectories. Many approaches reduce the degree of replication, e.g., by expecting more computational resources from participating nodes or additional trust assumptions. This seems particularly attractive in scenarios that already have a relatively strong degree of centralization and contractual agreements between participants (and their CSPs) and where the centralization tradeoff is acceptable to obtain better compatibility with legacy systems, cloud integration, and performance and cost structure. For permissionless blockchains, optimistic rollups introduce a simple approach to reduce the degree of replication at the cost of limited gains (no compression of data) and the need for some additional assumptions (which can be made quite weak through introducing incentives). Validity rollups, in contrast, can further reduce the amount of data that needs to be stored but are more complex to implement and put high resource requirements on the prover. Both approaches can achieve a separation of data availability and code execution and find individual optimizations for both challenges. Validity rollups have only recently emerged for general-purpose smart contract invocation and the design space for data availability layers with a reduced degree of replication (as a single honest party that provides the data is sufficient) has not yet been fully explored and implemented. Thus, both the maturity and complexity of these solution approaches may still be inhibiting for most enterprises to adopt these solutions. Notably, while there are many publications that explore the feasibility and performance gains of different scaling solutions from a theoretical computer science or engineering perspective, I found no research in the information systems domain that would give businesses decision support on which of these scaling solutions may be most appropriate to solve their problems.

III.4 Transparency

III.4.1 Problematization

Just like for the discussion of performance and operating costs, the replicated storage and execution of transactions is also the core reason why blockchain-based information systems exhibit excessive transparency (research paper 6). Indeed, "preserving privacy of participants and confidentially of their data has turned out to be a fundamental challenge" (Butijn et al., 2020, p. 19) in blockchain-based systems: Every blockchain node has full access to the append-only ledger and the corresponding state at any time, as well as all inputs, intermediary results, and outputs of computations associated with transaction processing. While initially cryptocurrencies like Bitcoin marketed the pseudonymity that they provide through their public keys (or hashes thereof) as a significant advantage as compared to the transparency of bank accounts verified in a KYC process from the perspective of a financial institution, it soon turned out that with sophisticated analyses, the linkability of transactions allows to de-anonymize many blockchain addresses and to associate the corresponding accounts with individuals or organizations (see, e.g., Béres et al., 2021; Biryukov and Tikhomirov, 2019). Individuals can to some extent choose freely whether they want to accept these privacy risks when engaging in payments or DeFi-related transactions or abstain from engaging in these activities. Yet, the situation is more daunting for organizations. First, their business model typically depends on engaging in data processing activities. Second, they are more likely to be identifiable through using their public key or blockchain address when engaging in contractual, regulated activities. Third, organizations need to meet various requirements, such as their customers' privacy expectations and regulation that imposes strict rules on how sensitive data must be managed, e.g., related to data protection and antitrust regulation (research paper 6; Toufaily et al., 2021). Thus far, related work has mainly focused on the tension between blockchain-based information systems and data protection regulation and in particular the European Union's GDPR, which is perceived as particularly strict (Haque et al., 2021). Indeed, the replication of all information processed on blockchains conflicts with the GDPR principles of data minimization, purpose limitation, and storage limitation (research paper 6). The practical immutability of blockchains further aggravates the situation, as the right to be forgotten cannot be implemented (Rieger et al., 2019; Schellinger et al., 2022).

In a systematic analysis of different areas of blockchain application, research paper 6 indeed identified many different types of sensitive information that have been suggested for processing in organizational blockchain-based systems. For instance, in blockchain-based value transfer, individuals' and businesses' revenues, expenses, balances, turnover, and business partners may be disclosed (research paper 6). For application fields that aim to achieve tamper-proof documentation, e.g., notarizing documents (European Commission, 2021) or historical data about cars (Zavolokina et al., 2020b) on a blockchain, the content and validity status of personal or otherwise sensitive documents or valuable information that could be sold for profit on a market is inevitably accessible on the blockchain. Furthermore, when using blockchain-based solutions for cross-organizational workflow management, such as in international logistics (TradeLens Collaboration, 2022) or medical supply chains (Mattke et al., 2019), the frequency and type of processes and business relationships between the organizations involved may be revealed. When used for digital identity management, personal information related to individuals, such as their name, address, phone number, health information, or other authorizations and achievements would be accessible publicly (permissionless blockchain) or to organizations not involved in the cause of data processing (permissioned blockchain). Finally, even if one considers machines such as algorithmic traders, renewable energy assets that decide which local flexibility markets to join, or autonomous cars or robots as main stakeholders in a *machine economy*, where blockchains are supposed to enable micropayments and autonomous asset exchanges (Jöhnk et al., 2021; Schweizer et al., 2020), these issues are still prevalent as the machines are typically the property of individuals or businesses, and so sensitive information associated with the robot can also be considered sensitive information associated with these entities.

Consequently, a significant body of research and practical implementations has tried to address the challenge of excessive transparency. As in the case of performance (Section III.3.1), permissioned blockchains have often been suggested as a solution to this issue (e.g., Pedersen et al., 2019). Indeed, permissioned blockchains can to some extent mitigate the data visibility issue by restricting node operation and, therefore, data visibility to only selected entities. However, this approach is not sufficient for any of the three above-mentioned scenarios: From a regulatory perspective, purpose limitation supposedly covers any unnecessary replication (research paper 6). Moreover, while "redactable blockchains" to implement the right to be forgotten seem more practical for permissioned than the permissionless settings (Ateniese et al., 2017; Deuber et al., 2019), corresponding solutions introduce many new problems, such as dynamic adjustments of backdoors to the set of nodes. Lastly, permissioned blockchains are also more vulnerable to data breaches than a centralized system, as a fault or attack on the node run by the organization with the weakest security measures is sufficient (Schlatt et al., 2022b). In sum, storing sensitive data on a fully transparent layer is arguably a poor design choice that should be addressed in the first place.

To address these limitations of permissioned blockchains to address the excessive transparency of data processed on blockchains, certain popular permissioned blockchains like Hyperledger Fabric and Quorum have introduced the concept of "private transactions" (research paper 4; Consensys/GoQuorum, 2021). These private transactions involve distributing hashed or encrypted data to all nodes, with only specific nodes designated at the smart contract or transaction level being allowed to execute the transaction based on the original data. These selected nodes can obtain the original data through a peer-to-peer messaging layer (Hyperledger Fabric) or by retrieving encrypted data from the blockchain and decrypting it (Quorum). Similar approaches can be applied also in permissionless blockchains. By using encryption to distribute information only to the relevant parties in a blockchain-based interaction, the amount of information exposed is reduced without necessarily compromising trust and verifiability guarantees among the interacting parties. However, restricted access to on-chain information results in substantially reduced functionality for smart contracts. For instance, restricting information sharing and, thus, verifiability is not suitable for many interactions that involve digital assets: The whole network that interacts with this asset needs to verify the compliance with the basic transactional rules (e.g., the receiver's balance is increased by no more than the reduction on the sender's balance).

Processing encrypted or otherwise obfuscated data on a blockchain also introduces many challenges while providing unclear benefits: On the one hand, in the future, more powerful computers (both quantum and classical) may allow decrypting data persisted on blockchains that is state-of-the-art encrypted as of today (research paper 6). Since the data cannot be simply deleted from a blockchain, one also cannot simply perform periodic re-encryption to keep up with the latest tooling. On the other hand, blockchains can be used for proving data integrity without exposing sensitive information by simple and less vulnerable means, i.e., by storing its hash (Schellinger et al., 2022), which also allows further reducing corresponding costs when compressing many hashes in Merkle trees (Chod et al., 2020; Djamali et al., 2021). In contrast, encrypted data consumes at least as much space as non-encrypted data. Even more, the value added of encrypted data for information *processing* on a blockchain is not clear, as smart contracts in general cannot make use of encrypted data: While algorithms can compute on encrypted data ("fully homomorphic encryption"), as of today they are still considered too computationally expensive to be suitable for deployment on a blockchain (Garrido et al., 2022). Consequently, encrypted data provides no obvious benefits for blockchain-based processes using smart contracts but substantial risks. Lastly, because of the aforementioned risks and the inefficiency and costs related to replicated processing, using blockchains and encryption for simple bilateral data exchange seems inferior to direct interactions using REST-APIs and encrypted traffic (as in the HTTPS protocol). This approach is a common practice and is further pursued and standardized in the context of business-to-business interactions, as in the European GAIA-X consortium and in particular its *dataspace connectors* (Otto, 2022). Off-chain data exchange via bilateral communication is also more appropriate from another perspective, as it avoids the need for costly storage on a blockchain (see Section III.3.1).

On the other hand, such bilateral exchange cannot address workflows that were believed to represent the core value proposition of blockchain networks, such as the transfer of ownership of digital assets or the verifiable tracking of manufactured goods through the supply chain. In sum, there seems to be a profound tradeoff between the completeness of information recorded on a blockchain, which implies excessive visibility, and the scope of verifiable and trusted interactions that can be facilitated by smart contracts (research paper 6; Kannengießer et al., 2020). Similar to the issues concerning performance, a permissioned blockchain can mitigate excessive transparency to some extent because it restricts data visibility to a set of well-defined entities. However, even permissioned blockchains draw their core value from uniting competitors on the same digital infrastructure (Fridgen et al., 2018c). Consequently, sharing sensitive information by default with all entities participating in a permissioned blockchain can already be considered excessive (research paper 8).

III.4.2 Solution approaches

As discussed in Section III.4.1, data that is processed in a smart contract is by definition replicated on every blockchain node. The main reason for the replicated storage of data and the replicated execution of transactions such as smart contract calls is the need for verifiability ("safety") to ensure that all honest nodes have synchronized state. Consequently, the only approach to mitigate the transparency issue is to reduce the amount of information processing on the blockchain and smart contract level. There are two main strategies to do so, depending on whether organizations seek only blockchain-based data storage for obtaining data integrity guarantees or also smart contract execution, which requires additional computational integrity guarantees (Schellinger et al., 2022).

First, if no verifiability of computations is required, then sensitive data can be removed from the blockchain relatively easily. For example, if the integrity of data needs to be demonstrated to third parties, storing the hash value corresponding to the data on the blockchain and the data separately off-chain (with a sufficient number of backups) may be sufficient for most applications that involve tamper-resistant data (Schellinger et al., 2022). This approach also comes with much smaller challenges regarding access control, data protection, and integration (Rieger et al., 2021). In scenarios in which the provenance of the data is external and, thus, the authenticity of the data relies on a third party's digital signatures in the first place, blockchain-based storage for verifiability may not be necessary at all, as the digital signature also facilitates tamper-evidence. The approach based on digital signatures has lately been adopted by many digital identity initiatives, considering that for verifiable identity-related documents, a trust relationship with the issuer of these documents is required anyway (Rieger et al., 2021; Toubiana et al., 2022). This perspective suggests that the role of blockchain for digital identity management is relatively limited (research paper 7 and 8). Many endeavors in the public and the private sectors that implement such digital certifications ("verifiable credentials") have moved to an architecture in which blockchain only takes a role in the recording of information that is intended to be public, such as cryptographic keys for trusted entities that issue (sign) corresponding attestations (research paper 7; Sedlmeir et al., 2021), but not the potentially sensitive identity attributes themselves. In this context, it is important to note that blockchain is by far not the only way to validate data that is stored in a decentralized way (research paper 7). The Internet has done so since the 1990s based on digital signatures and public key infrastructures. During the pandemic, machine-verifiable COVID-19 vaccination credentials have been implemented purely on this basis without using blockchain technology at all, for instance, in the European Union (Rieger et al., 2021). Consequently, interactions that focus on sharing verifiable, sensitive data, ranging from personal and health data to meta-data and transactional data involved in business interactions, should and can be moved off the blockchain (Platt et al., 2021a). In contrast, the availability of such a separate layer for the verifiable, bilateral, and confidential exchange of sensitive information may even help to separate sensitive from non-sensitive data in some blockchain applications and to move the interactions involving sensitive data off the blockchain and, thus, help address the transparency issues of blockchain technology (research paper 6).

The more complex scenario for data visibility involves blockchain-based transaction processing. To address this issue, a crucial observation is that verifiability does not necessarily require full transparency that facilitates the re-execution of a transaction. For example, the verification of a simple blockchain-based payment typically involves the verification of the sender's digital signature (an authorization check) and the verification that the sender has enough funds to cover the transaction (Ben-Sasson et al., 2014). Without these verifications, any holder of the corresponding currency can incur negative consequences because someone else could spend their funds (if authorization checks are not performed) or the total supply of the currency could be increased, reducing the value of other holders' assets (if checks that the sender's funds are sufficient are not performed). Yet, neither the sender's nor the receiver's identity nor the transferred amount per se play an important role to third parties. One core idea behind many privacy solutions on blockchains is hence the decoupling of the verifiability of a computation (here: the two checks) from the actual data and computation. This can be achieved using secure computing techniques and in particular with TEEs or ZKPs (Garrido et al., 2022).

Besides providing attestations for the correctness of code executed in them, TEEs can "operate on encrypted data". This does not mean that they directly compute on encrypted data, as in the case of FHE, but rather that they can use the cryptographic keys that they securely store to decrypt encrypted inputs, use this data in a computation, encrypt the result inside the TEE and return the result; including the attestation for correct execution that allows all blockchain nodes to verify the correctness of the result of the operation. TEEs provide these operations with relatively high performance (e.g., Cheng et al., 2019) and do not involve a need to develop new dedicated cryptographic tools. As in the case of scaling, an "idealized" TEE would therefore be a silver bullet for solving the transparency challenge, but because of their aforementioned problems in practice, alternative cryptographic solutions have become more popular.

ZKPs are arguably the most prominent representative of these cryptographic approaches. They can replace the trust in the correct execution of hardware with trust in mathematics and cryptography and are therefore considered a more secure alternative to TEEs for providing this specific functionality. ZKPs are already in use, for instance, to prove the integrity of medical supply chains in a blockchain-based IT system and, therefore, to improve the detection of fake medicals (Mattke et al., 2019). Moreover, they can also address the enforcement of rules, such as the invariance of total supply and the auditability of workflows, without disclosing the underlying, potentially sensitive data (Jeong et al., 2023). While ZKPs technologies are already in productive use in some blockchain-based applications (Ben-Sasson et al., 2014; Mattke et al., 2019), and they have matured rapidly in the last years (Ben-Sasson et al., 2019; Bootle et al., 2020; Thaler, 2020), they increase complexity significantly. Moreover, despite recent efficiency improvements by several orders of magnitude, they are still subject to severe performance limitations: While their verification on a blockchain is in general very fast – at least for a particular subset of ZKPs, namely zk-SNARKs (their succinctness can even improve performance, see Section III.3.2) – the "prover" that executes a transaction locally and provides evidence for the correct execution incurs substantial overhead. Consequently, today general-purpose ZKPs are only applicable for relatively simple computations that take less than a second to perform on a common CPU as of today (Thaler, 2022). Moreover, ZKPs are also conceptually more limited than TEE-based approaches because they require access to all data underlying the computation for which the correct execution is to be proven (research paper 6; Buterin, 2014). Moreover, many businesses do not have the expertise in cryptography and engineering to implement such solutions. A promising strategy taken by successful solutions like MediLedger (Mattke et al., 2019) is to slightly adapt innovative ZKP-based solutions in cryptocurrencies, which drive innovation in this domain through the available talent and funding and provide open-source and often well-audited implementations. Finally, even if a good solution for the on-chain verification of transactions is found, it remains an open question how sensitive data can be exchanged between authorized stakeholders in a standardized way that can readily be combined with a blockchain-based solution. Such data exchange would require fine-granular access management implemented across organizations and, consequently, a uniform way to issue and verify authorizations and permissions in a system that does not re-introduce new single points of failure. The aforementioned digital identity solutions may be a suitable building block for such systems.

Besides TEE- and ZKP-based approaches, there are also other advanced cryptographic concepts, such as MPC and FHE that can contribute to addressing the transparency challenge. Both approaches allow different organizations to perform joint verifiable operations on data that does not need to be disclosed to a smart contract. As such, MPCs and FHEs can handle some additional workloads where no entity has access to all data underlying a computation but seem less mature because they are communication- and data-intensive, respectively, both of which represent very scarce and expensive resources on blockchains (research paper 6; Garrido et al., 2022). Consequently, these concepts are particularly challenging to use when performance improvements are required at the same time.

In sum, issues related to excessive transparency will arguably require sophisticated combinations of off-chain bilateral exchange of sensitive data with the selective use of privacy-enhancing cryptographic techniques like ZKPs. It is also important to highlight that many of these privacy-enhancing technologies involve higher amounts of on-chain computation than the corresponding task without considering privacy. This is even true for zk-SNARKs despite their succinctness in some cases, such as very simple digital asset transfers, as their verification is still more complex than the verification of a single digital signature. Consequently, researchers and projects are exploring also the combination of scaling and privacy solutions, and in particular the combination of zk-SNARKs for addressing both issues at the same time in what is sometimes called zk^2-rollups. One popular example is Aztec Connect[14]. Combining scaling and privacy solutions puts novel challenges on engineering in multiple dimensions. For instance, it has accelerated research on solutions such as SNARK recursion that allows verifying the correct execution of the verification of one or several (zk-)SNARK inside a SNARK (Kothapalli and Setty, 2023). It has also inspired research on the combination of ZKPs with MPC to achieve more efficient MPC protocols that are secure against arbitrary faults (Ozdemir and Boneh, 2022). However, moving most of the data previously operated on a blockchain off-chain and mainly storing cryptographic proofs for the correct operation also introduces new challenges regarding the composability of smart contracts, which seems to be one of the key benefits of blockchain use in some sectors such as DeFi. Consequently, it is conceivable that despite further progress in the performance of these solutions and the reduction of complexity for software developers who implement smart contracts, organizations will require decision support on how to combine and use all these different building blocks for confidential blockchain interactions. In particular, taking the discussion of complex, often cryptographic, tooling to reduce the degree of excessive data visibility on blockchains aside, it seems worthwhile to provide organizations with guidance on which kind of data should and should not be stored on blockchains before engaging in solution design.

[14] See https://aztec.network/.

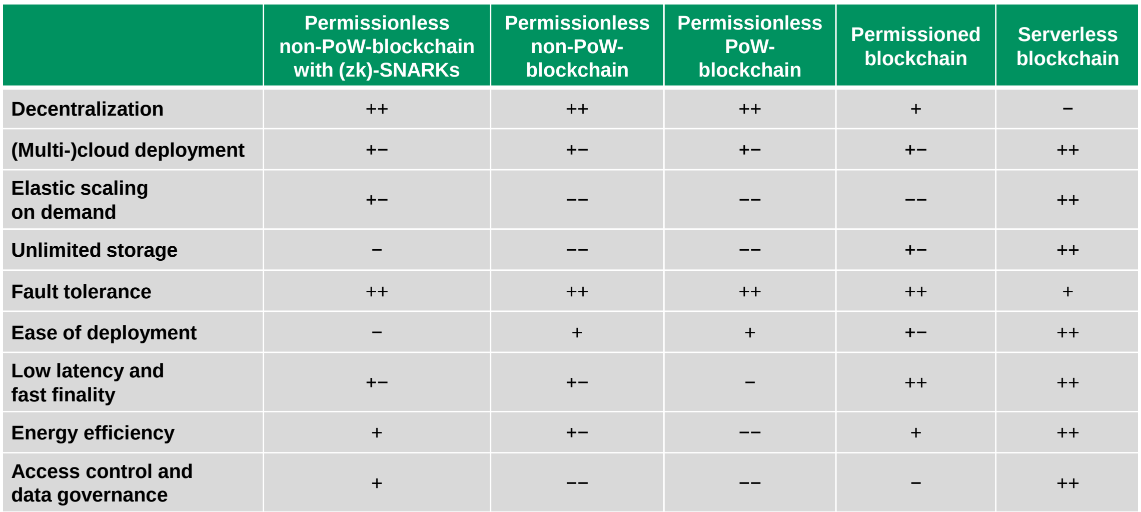
IV Conclusion

Blockchain technology has attracted significant attention in research and practice. Various companies and public-sector institutions have explored the opportunities of implementing blockchain-based information systems to improve their processes. This doctoral thesis argues that some technical challenges of blockchain technology have thus far not received the appropriate weight in information systems research. Many information systems and application-oriented publications have predominately pointed to management-related issues or put a narrow focus on sustainability issues when discussing potential issues of blockchain use from a technical side. I demonstrate that the energy consumption of blockchain technology has already been well assessed and that there is a variety of mature solutions – namely, non-PoW blockchains – with relatively low energy needs (research papers 1 and 2). In contrast to PoW blockchains, the energy consumption of these non-PoW blockchains will also further decrease through future advancements in hardware and software. Moreover, this dissertation showcases that limitations and tradeoffs inherent to blockchain systems according to their core characteristic, namely replicated information processing, may have been underestimated by organizations and research that aims to guide them in adopting blockchain technology and designing corresponding solutions. I refer particularly to (1) challenges related to low performance and high operating costs on both permissionless blockchains (Ilk et al., 2021) and permissioned blockchains (research papers 3, 4, and 5) and (2) the processing of sensitive information beyond discussions of the GDPR and its right to be forgotten (research papers 6, 7, and 8). While these aspects have received substantial attention in computer science and inspired the rapid development of new concepts in academia and their implementation in practice (e.g., in verifiable computation and particularly (zk-)SNARKs), there has been surprisingly little research on which of these often highly complex solutions are most suitable for organizations to adopt, particularly if the organizations' core business is outside IT system design and implementation.

Many organizations that try to scale their blockchain applications to a larger scope or a higher number of participating entities are only now realizing the significant challenges blockchains pose in terms of performance and data visibility aspects. Thus far, they may have hoped that they can resolve these issues by using permissioned blockchain designs. This doctoral thesis emphasizes that these blockchain frameworks are neither necessary to address high energy consumption nor are they sufficient to resolve the performance and excessive transparency challenges entirely. Considering a plethora of publications that propose permissioned blockchain designs for organizations' IT systems, this dissertation hence suggests that an intensified academic discussion of the usefulness of permissioned blockchains is necessary. The case for permissioned blockchains has been contested by some stakeholders already quite early when the first conceptual solutions for improving performance and data visibility aspects were emerging for permissionless blockchains (Brody, 2019). Accordingly, future research may also challenge existing decision support, such as decision trees for guidance on blockchain use (Pedersen et al., 2019; Wüst and Gervais, 2018), and supplement it with more fine-granular steps that take into account the types of digital interactions and the sensitivity of related data.

As the scope of this doctoral thesis is relatively broad, it is naturally subject to several limitations. First, while I provide novel quantifications and assessments of the severity of different technical challenges and supplement them with rich empirical evidence and anecdotal examples, this dissertation does not include a validated ranking of the technical challenges from the perspective of stakeholders. This doctoral thesis also leaves an empirical assessment of the complexity of implementing and adopting the proposed solution approaches in organizations for future research. In particular, owing to the novelty of many of the technical tools discussed in this dissertation, evaluations of the solutions that I have designed and proposed relied mostly on experiments to obtain performance metrics, logical reasoning, and experts' assessments. To further substantiate the solutions I proposed, it is indispensable to also assess them by observing their operation in productive systems at scale.

According to my analysis, serverless blockchains can fully address most of the technical requirements organizations have, but the corresponding high degree of centralization makes them applicable only in very specific scenarios. Moreover, research in cryptography has made rapid advances in addressing both performance and privacy challenges, with zk-SNARKs providing substantial scalability improvements for permissionless blockchains and facilitating a higher degree of information disclosure control for both types of blockchain networks. However, there is thus far not a mature uniform solution to the performance and transparency challenges, and most concepts further contribute to the complexity that blockchain-based systems already suffer from in their simple off-the-shelf form. Figure 6 summarizes to which extent solution archetypes discussed in this doctoral thesis satisfy the requirements organizations pose on their blockchain-based IT systems identified in research paper 5. The solution archetypes included in this comparison are serverless blockchains, permissioned blockchains, permissionless PoW blockchains, permissionless PoS blockchains, and permissionless non-PoW blockchains that use SNARKs



	Permissionless non-PoW-blockchain with (zk)-SNARKs	Permissionless non-PoW-blockchain	Permissionless PoW-blockchain	Permissioned blockchain	Serverless blockchain
Decentralization	++	++	++	+	−
(Multi-)cloud deployment	+−	+−	+−	+−	++
Elastic scaling on demand	+−	−−	−−	−−	++
Unlimited storage	−	−−	−−	+−	++
Fault tolerance	++	++	++	++	+
Ease of deployment	−	+	+	+−	++
Low latency and fast finality	+−	+−	−	++	++
Energy efficiency	+	+−	−−	+	++
Access control and data governance	+	−−	−−	−	++

Figure 6: Comparison to which degree different paradigms of blockchain design address core technical challenges organizations face when adopting the technology.

for validity rollups to improve maximum throughput and zk-SNARKs for reducing information disclosure without compromising verifiability.

From a practitioner's perspective, this dissertation makes a contribution by illustrating to decision-makers and information systems engineers in organizations that there are many blockchain designs available that avoid energy consumption issues (research papers 1 and 2), and that they should rather focus on identifying solutions that allow addressing performance and cost limitations (research papers 3 and 4) or provide fine-granular control of the disclosure of sensitive information (research paper 6) when designing and implementing blockchain-based solutions. This doctoral thesis also provides a comprehensive analysis of the extent to which electricity consumption, performance and cost aspects, and transparency challenges affect blockchain applications in organizations. It hence gives guidance on how solutions to these challenges can be identified and deployed effectively. Some publications included in this dissertation design and evaluate specific solutions, particularly in the context of the decentralized, verifiable exchange of sensitive personal information, and discuss the corresponding tradeoffs and role of blockchain-based designs. This dissertation also explores how the substantially more restricted use of blockchain in applications involving the exchange of machine-verifiable identity information (see research papers 7 and 8) can help address issues related to the transparency of blockchain-based information processing. These publications also emphasize the relevance of taking a multi-disciplinary perspective that considers not only engineering aspects but also business requirements and regulatory constraints from the start rather than trying to solve these challenges retrospectively when scaling the blockchain solution to additional stakeholders and processes. Organizations hoping for more mature solutions in the coming years are well advised to consider not only management-related aspects of blockchain deployments but also to prepare for advanced solutions by discussing which kind of data should be stored and processed on the blockchain layer and for which processes and statements public verifiability is essential. Furthermore, I advise organizations to investigate whether simpler technologies that provide a subset of the capabilities of blockchains, such as public key infrastructures and digital signatures, are sufficient for meeting their requirements, particularly when they are more focused on the exchange of data than on the exchange of value or the verifiability of cross-organizational digital processes. Such technologies are abundant on the Web already today and can also provide high degrees of decentralization and integrity without being exposed to the performance, cost, complexity, and data visibility issues related to blockchain solutions.

I conclude that sustainability aspects are hardly problematic for blockchain technology in general but rather an undue extrapolation from Bitcoin's high electricity needs. Performance and privacy aspects seem more fundamental and challenging to solve for organizations. In particular, the permissioned blockchains that have been developed to address these issues often do not provide a sufficient solution unless their degree of decentralization is even further increased in the form of "serverless blockchains". On the other hand, promising – in particular cryptographic – concepts exist for addressing them also for highly decentralized, permissionless blockchain networks, but most corresponding solutions are still incomplete or complex to design. Considerable work remains to be done to communicate the technical specificities and tradeoffs of these solutions to decision-makers and to make them easy to deploy and operate on a large scale. System architects and information systems researchers can benefit from keeping a close eye on related academic research in computer science (e.g. Bünz, 2023) and practical deployments (e.g., in the Ethereum ecosystem) to identify potential technical challenges early and to transfer related solutions, e.g., ones based on zk-SNARKs, to their business applications.

V Acknowledgment of previous and related work

In my academic and project work, I collaborated with many colleagues at the FIM Research Center, the Branch Business & Information Systems Engineering of the Fraunhofer Institute for Applied Information Technology (FIT), the University of Bayreuth, the Interdisciplinary Center for Security, Reliability and Trust at the University of Luxembourg, the University of Augsburg, the Augsburg University of Applied Sciences, and the Frankfurt University of Applied Sciences. Many of the publications I co-authored were inspired or informed by these colleagues' or their supervisors' previous publications. To acknowledge this indispensable contribution to my dissertation, I present how my work builds on and connects to their work below.

Foundational work on technical and socio-economic properties of blockchain technology and related application areas was laid by Arnold et al. (2019), Fridgen et al. (2018a), Fridgen et al. (2018b), Fridgen et al. (2018c), Fridgen et al. (2018d), Schlatt et al. (2016), Schütte et al. (2018), and Schweizer et al. (2017). Together with several of these colleagues, I frequently discussed the benefits blockchains can provide for coordinating cross-organizational workflows and core technical properties of such systems, including energy consumption and performance aspects. Some of these discussions were condensed in a report for the German Ministry for Transport and Digitalization (Fridgen et al., 2019). A refinement of the energy consumption characteristics discussed in this report, combined with the identification of energy savings potentials, e.g., through blockchain applications in the energy sector (Albrecht et al., 2018; Strüker et al., 2019; Utz et al., 2018) and supply chain management (Guggenberger et al., 2020b), revealed the research gap underlying research paper 1. This publication also connects to another direction of the research groups that covers myths about blockchain technology, e.g., regarding the degree of decentralization (Barbereau et al., 2022; Barbereau et al., 2023) and specific tradeoffs in token design (Drasch et al., 2020). Further ongoing discussions in the research group about the sustainability of blockchain technology – to which Fridgen et al. (2021a) also contributed – motivated research paper 2.

Besides energy consumption, the research group's members focused on several other challenges in the context of blockchain adoption in organizations. These research activities also cover some technical perspectives like attack vectors on blockchain networks (Schlatt et al., 2022a) but mainly focus on managerial perspectives, such as the works by Guggenberger et al. (2021) and Feulner et al. (2022). My research on performance aspects of blockchain technology – with a focus on permissioned blockchain networks – builds on the identified demand for enterprise blockchains with higher performance guarantees and complement the identification of technical capabilities and challenges through systematic performance analyses in research papers 3 and 4. This research direction also laid the foundation for analyzing serverless blockchains as an alternative, more centralized approach to enterprise blockchains that focuses on exceptionally high performance requirements or established access control mechanisms in organizations' cloud infrastructures (see research paper 6).

Through structuring the different application areas for blockchain-based information systems my colleagues and co-authors have explored, such as finance and digital assets ("tokens") (Bachmann et al., 2021; Regner et al., 2019; Schweizer et al., 2020; Sunyaev et al., 2021), supply chain management and trade finance (Fridgen et al., 2021b; Guggenberger et al., 2020b), machine economy (Jöhnk et al., 2021; Schweizer et al., 2020), e-government and services in federated environments (Amend et al., 2021b; Hoess et al., 2023; Roth et al., 2022a; Roth et al., 2023), as well as cross-organizational digital processes in general (Amend et al., 2021a; Fridgen et al., 2018c), I collected many examples for potentially sensitive information that may be exchanged in the context of blockchain applications. These publications hence pointed to the corresponding challenges and potential solutions in research paper 6. The research group started considering this aspect already early on in the context of GDPR requirements in public sector projects that involve personal data (Guggenmos et al., 2020; Rieger et al., 2019). These activities opened up questions on the relationship between blockchain technology and digital identities (Guggenberger et al., 2020a; Guggenmos et al., 2018; Lockl et al., 2018). We first provided an overview of approaches to digital identity and in particular the emerging self-sovereign identity paradigm in Sedlmeir et al. (2021) before we approached the topic more rigorously in research paper 8, where we developed nascent design principles on the role of blockchain for SSI. The remaining questions related to user centricity that the research group has thus far posed and tried to answer (Amard et al., 2022; Weigl et al., 2022b) also helped us with setting up subsequent research on SSI from a less technical perspective, as in user experience (Sartor et al., 2022) and user acceptance (Guggenberger et al., 2023) studies of digital identity wallets. Studying regulations that affect these emerging digital technologies also led the research group toward investigating general aspects of digital policies (Codagnone and Weigl, 2023; Weigl et al., 2022a) and tensions between them (Weigl et al., 2023).

VI References

451 Research (2019). *The carbon reduction opportunity of moving to Amazon Web Services*. URL: https://shorturl.at/ajmtK.

Adler, J. and M. Quintyne-Collins (2019). *Building scalable decentralized payment systems*. URL: https://arxiv.org/abs/1904.06441.

Aguiar, E. J. de, B. S. Faiçal, B. Krishnamachari, and J. Ueyama (2020). "A survey of blockchain-based strategies for healthcare". In: *ACM Computing Surveys* 53 (2). DOI: 10.1145/3376915.

Albrecht, S., S. Reichert, J. Schmid, J. Strüker, D. Neumann, and G. Fridgen (2018). "Dynamics of blockchain implementation – a case study from the energy sector". In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3527–3536. DOI: 10.24251/hicss.2018.446.

Aldweesh, A. (2023). "BlockTicket: A framework for electronic tickets based on smart contract". In: *Plos One* 18 (4), e0284166. DOI: 10.1371/journal.pone.0284166.

Alt, R. (2020). "Electronic markets on blockchain markets". In: *Electronic Markets* 30 (2), pp. 181–188. DOI: 10.1007/s12525-020-00428-1.

Alt, R. and E. Wende (2020). "Blockchain technology in energy markets – an interview with the European Energy Exchange". In: *Electronic Markets* 30 (2), pp. 325–330. DOI: 10.1007/s12525-020-00423-6.

Amard, A., A. Hoess, T. Roth, G. Fridgen, and A. Rieger (2022). "Guiding refugees through European bureaucracy: Designing a trustworthy mobile app for document management". In: *Proceedings of the International Conference on Design Science Research in Information Systems and Technology*. Springer, pp. 171–182. DOI: 10.1007/978-3-031-06516-3_13.

Amend, J., M. Federbusch, G. Fridgen, F. Köhler, A. Rieger, V. Schlatt, J. Sedlmeir, A. Stohr, and C. van Dun (2021a). *Digitization of certification processes in the asylum procedure by means of digital identities: A feasibility study by Germany's Federal Office for Migration and Refugees*. URL: https://orbilu.uni.lu/bitstream/10993/48332/1/blockchain-whitepaper-2021.pdf.

Amend, J., G. Fridgen, A. Rieger, T. Roth, and A. Stohr (2021b). "The evolution of an architectural paradigm – using blockchain to build a cross-organizational enterprise service bus". In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 4301–4310. DOI: 10.24251/hicss.2021.522.

Amend, J., J. Kaiser, L. Uhlig, N. Urbach, and F. Völter (2021c). "What do we really need? A systematic literature review of the requirements for blockchain-based e-government services”. In: *Innovation Through Information Systems: Volume I: A Collection of Latest Research on Domain Issues*, pp. 398–412. DOI: 10.1007/978-3-030-86790-4_27.

Andoni, M., V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum, and A. Peacock (2019). “Blockchain technology in the energy sector: A systematic review of challenges and opportunities”. In: *Renewable and Sustainable Energy Reviews* 100, pp. 143–174. DOI: 10.1016/j.rser.2018.10.014.

Androulaki, E. et al. (2018). “Hyperledger Fabric: A distributed operating system for permissioned blockchains”. In: *Proceedings of the 13th EuroSys Conference*. DOI: 10.1145/3190508.3190538.

Angelis, S. de, L. Aniello, F. Lombardi, A. Margheri, and V. Sassone (2017). *PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain*. URL: https://eprints.soton.ac.uk/415083/2/itasec18_main.pdf.

Arnold, L., M. Brennecke, P. Camus, G. Fridgen, T. Guggenberger, S. Radszuwill, A. Rieger, A. Schweizer, and N. Urbach (2019). “Blockchain and initial coin offerings: blockchain’s implications for crowdfunding”. In: *Business Transformation through Blockchain*. Springer, pp. 233–272. DOI: 10.1007/978-3-319-98911-2_8.

Arnosti, N. and S. M. Weinberg (2022). “Bitcoin: A natural oligopoly”. In: *Management Science* 68 (7). DOI: 10.1287/mnsc.2021.4095.

Arooj, A., M. S. Farooq, and T. Umer (2022). “Unfolding the blockchain era: Timeline, evolution, types and real-world applications”. In: *Journal of Network and Computer Applications* 207, p. 103511. DOI: 10.1016/j.jnca.2022.103511.

Ateniese, G., B. Magri, D. Venturi, and E. Andrade (2017). “Redactable blockchain – or – rewriting history in Bitcoin and friends”. In: *European Symposium on Security and Privacy*. IEEE, pp. 111–126. DOI: 10.1109/eurosp.2017.37.

AWS (2021). *Amazon EC2 pricing*. Amazon Web Services. URL: https://aws.amazon.com/ec2/pricing/on-demand/?nc1=h_ls.

Bachmann, N. M., B. Drasch, G. Fridgen, M. Miksch, F. Regner, A. Schweizer, and N. Urbach (2021). “Tarzan and chain: Exploring the ICO jungle and evaluating design archetypes”. In: *Electronic Markets*. DOI: 10.1007/s12525-021-00463-6.

Bagaria, V., A. Dembo, S. Kannan, S. Oh, D. Tse, P. Viswanath, X. Wang, and O. Zeitouni (2022). “Proof-of-stake longest chain protocols: Security vs predictability”. In: *Proceedings of the Workshop on Developments in Consensus*. ACM, pp. 29–42. DOI: 10.1145/3560829.3563559.

Bagaria, V., S. Kannan, D. Tse, G. Fanti, and P. Viswanath (2019). “Prism: Deconstructing the blockchain to approach physical limits”. In: *Proceedings of the 26th ACM SIGSAC*

Conference on Computer and Communications Security, pp. 585–602. DOI: 10.1145/3319535.3363213.

Baliga, A., N. Solanki, S. Verekar, A. Pednekar, P. Kamat, and S. Chatterjee (2018a). "Performance characterization of Hyperledger Fabric". In: *Crypto Valley Conference on Blockchain Technology*. IEEE, pp. 65–74. DOI: 10.1109/cvcbt.2018.00013.

Baliga, A., I. Subhod, P. Kamat, and S. Chatterjee (2018b). *Performance evaluation of the Quorum blockchain platform*. URL: http://arxiv.org/abs/1809.03421.

Barbereau, T., R. Smethurst, O. Papageorgiou, A. Rieger, and G. Fridgen (2022). "DeFi, not so decentralized: The measured distribution of voting rights". In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 6043–6052. DOI: 10.24251/hicss.2022.734.

Barbereau, T., R. Smethurst, O. Papageorgiou, J. Sedlmeir, and G. Fridgen (2023). "Decentralised finance's timocratic governance: The distribution and exercise of tokenised voting rights". In: *Technology in Society* 73, p. 102251. DOI: 10.1016/j.techsoc.2023.102251.

Barr, J. (2019). *Amazon Prime day 2019 – Powered by AWS*. URL: https://aws.amazon.com/blogs/aws/amazon-prime-day-2019-powered-by-aws/.

Bauer, I., J. Parra-Moyano, K. Schmedders, and G. Schwabe (2022). "Multi-party certification on blockchain and its impact in the market for lemons". In: *Journal of Management Information Systems* 39 (2), pp. 395–425. DOI: 10.1080/07421222.2022.2063555.

Bauer, I., L. Zavolokina, and G. Schwabe (2019). "Is there a market for trusted car data?" In: *Electronic Markets* 30 (2), pp. 211–225. DOI: 10.1007/s12525-019-00368-5.

Beck, R., M. Avital, M. Rossi, and J. B. Thatcher (2017). "Blockchain technology in business and information systems research". In: *Business & Information Systems Engineering* 59 (6), pp. 381–384. DOI: 10.1007/s12599-017-0505-1.

Beck, R., C. Müller-Bloch, and J. L. King (2018). "Governance in the blockchain economy: A framework and research agenda". In: *Journal of the Association for Information Systems* 19 (10), pp. 1020–1034. DOI: 10.17705/1jais.00518.

Behnke, K. and M. Janssen (2020). "Boundary conditions for traceability in food supply chains using blockchain technology". In: *International Journal of Information Management* 52, p. 101969. DOI: 10.1016/j.ijinfomgt.2019.05.025.

Ben-Sasson, E., I. Bentov, Y. Horesh, and M. Riabzev (2019). "Scalable zero knowledge with no trusted setup". In: *Annual International Cryptology Conference*, pp. 701–732. DOI: 10.1007/978-3-030-26954-8_23.

Ben-Sasson, E., A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza (2014). “Zerocash: Decentralized anonymous payments from Bitcoin”. In: *Symposium on Security and Privacy*. IEEE, pp. 459–474. DOI: 10.1109/SP.2014.36.

Ben-Sasson, E., A. Chiesa, D. Genkin, E. Tromer, and M. Virza (2013). “SNARKs for C: Verifying program executions succinctly and in zero knowledge”. In: *Annual Cryptology Conference*. Springer, pp. 90–108. DOI: 10.1007/978-3-642-40084-1_6.

Béres, F., I. A. Seres, A. A. Benczúr, and M. Quintyne-Collins (2021). “Blockchain is watching you: Profiling and deanonymizing Ethereum users”. In: *International Conference on Decentralized Applications and Infrastructures*. IEEE, pp. 69–78. DOI: 10.1109/DAPPS52256.2021.00013.

Bernstein, D. (2014). “Containers and cloud: From LXC to Docker to Kubernetes”. In: *IEEE Cloud Computing* 1 (3), pp. 81–84. DOI: 10.1109/MCC.2014.51.

Biryukov, A. and D. Feher (2020). “ReCon: Sybil-resistant consensus from reputation”. In: *Pervasive and Mobile Computing* 61, p. 101109. DOI: 10.1016/j.pmcj.2019.101109.

Biryukov, A. and S. Tikhomirov (2019). “Deanonymization and linkability of cryptocurrency transactions based on network analysis”. In: *European Symposium on Security and Privacy*. IEEE, pp. 172–184. DOI: 10.1109/eurosp.2019.00022.

Bonneau, J., A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten (2015). “SoK: Research perspectives and challenges for Bitcoin and cryptocurrencies”. In: *Symposium on Security and Privacy*. IEEE, pp. 104–121. DOI: 10.1109/SP.2015.14.

Bons, R. W. H., J. Versendaal, L. Zavolokina, and W. L. Shi (2020). “Potential and limits of blockchain technology for networked businesses”. In: *Electronic Markets* 30 (2), pp. 189–194. DOI: 10.1007/s12525-020-00421-8.

Bootle, J., A. Chiesa, and S. Liu (2020). *Zero-knowledge succinct arguments with a linear-time prover*. URL: https://eprint.iacr.org/2020/1527.pdf.

Brody, P. (2019). *How public blockchains are making private blockchains obsolete*. URL: https://www.ey.com/en_gl/innovation/how-public-blockchains-are-making-private-blockchains-obsolete.

Buchman, E. (2016). *Tendermint: Byzantine fault tolerance in the age of blockchains*. URL: https://atrium.lib.uoguelph.ca/xmlui/bitstream/handle/10214/9769/Buchman_Ethan_201606_MAsc.pdf?ref=hackernoon.com.

Budish, E. (2018). “The economic limits of Bitcoin and the blockchain”. In: *NBER Working Paper Series*. DOI: 10.3386/w24717.

Bumblauskas, D., A. Mann, B. Dugan, and J. Rittmer (2020). “A blockchain use case in food distribution: Do you know where your food has been?” In: *International Journal of Information Management* 52, p. 102008. DOI: 10.1016/j.ijinfomgt.2019.09.004.

Bünz, B. (2023). “Improving the privacy, scalability, and ecological impact of blockchains”. PhD thesis. Stanford University. URL: https://stacks.stanford.edu/file/druid:pz524zp4725/thesis-no-copyright-augmented.pdf.

Bünz, B., L. Kiffer, L. Luu, and M. Zamani (2020). “Flyclient: Super-light clients for cryptocurrencies”. In: *Symposium on Security and Privacy*. IEEE, pp. 928–946. DOI: 10.1109/SP40000.2020.00049.

Buterin, V. (2014). *Secret sharing DAOs: The other crypto 2.0*. URL: https://blog.ethereum.org/2014/12/26/secret-sharing-daos-crypto-2-0/.

Buterin, V. (2020). *Why proof of stake*. URL: https://vitalik.ca/general/2020/11/06/pos2020.html.

Buterin, V. (2021). *The limits to blockchain scalability*. URL: https://vitalik.ca/general/2021/05/23/scaling.html.

Butijn, B.-J., D. A. Tamburri, and W.-J. van den Heuvel (2020). “Blockchains: A systematic multivocal literature review”. In: *ACM Computing Surveys* 53 (3). DOI: 10.1145/3369052.

Cambridge Centre for Alternative Finance (2020a). *Bitcoin network power*. URL: https://www.cbeci.org/.

Cambridge Centre for Alternative Finance (2020b). *Ethereum network power demand*. URL: https://ccaf.io/cbnsi/ethereum.

Carter, N. (2021). “How much energy does Bitcoin actually consume?” In: *Harvard Business Review* 5. URL: https://hbr.org/2021/05/how-much-energy-does-bitcoin-actually-consume.

Castro, M., B. Liskov, et al. (1999). “Practical Byzantine fault tolerance”. In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation*, pp. 173–186. URL: https://pmg.csail.mit.edu/papers/osdi99.pdf.

Catalini, C. and J. S. Gans (2020). “Some simple economics of the blockchain”. In: *Communications of the ACM* 63 (7), pp. 80–90. DOI: 10.1145/3359552.

Centobelli, P., R. Cerchione, P. Del Vecchio, E. Oropallo, and G. Secundo (2022). “Blockchain technology for bridging trust, traceability and transparency in circular supply chain”. In: *Information & Management* 59 (7), p. 103508. DOI: 10.1016/j.im.2021.103508.

Chanson, M., A. Bogner, D. Bilgeri, E. Fleisch, and F. Wortmann (2019). “Blockchain for the IoT: Privacy-preserving protection of sensor data”. In: *Journal of the Association for Information Systems* 20 (9), pp. 1274–1309. DOI: 10.17705/1jais.00567.

Cheng, R., F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, and D. Song (2019). “Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts”. In: *European Symposium on Security and Privacy*. IEEE, pp. 185–200. DOI: 10.1109/EuroSP.2019.00023.

Chod, J., N. Trichakis, G. Tsoukalas, H. Aspegren, and M. Weber (2020). “On the financing benefits of supply chain transparency and blockchain adoption”. In: *Management Science* 66 (10), pp. 4378–4396. DOI: 10.1287/mnsc.2019.3434.

Codagnone, C. and L. Weigl (2023). “Leading the charge on digital regulation: The more, the better, or policy bubble?” In: *Digital Society* 2.1. DOI: 10.1007/s44206-023-00033-7.

Consensys/GoQuorum (2021). *Private transaction lifecycle*. URL: https://docs.goquorum.consensys.net/en/stable/Concepts/Privacy/PrivateTransactionLifecycle/.

Costantinides, P., G. Parker, and O. Henfridsson (2018). “Platforms and infrastructures in the digital age”. In: *Information Sytems Research* 29 (2), pp. 381–400. DOI: 10.1287/isre.2018.0794.

Dabbagh, M., M. Sookhak, and N. S. Safa (2019). “The evolution of blockchain: A bibliometric study”. In: *IEEE Access* 7, pp. 19212–19221. DOI: 10.1109/ACCESS.2019.2895646.

Daian, P., S. Goldfeder, T. Kell, Y. Li, X. Zhao, I. Bentov, L. Breidenbach, and A. Juels (2020). “Flash Boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability”. In: *Symposium on Security and Privacy*. IEEE, pp. 910–927. DOI: 10.1109/sp40000.2020.00040.

De Filippi, P. and A. Wright (2018). *Blockchain and the law: The rule of code*. Harvard University Press.

de Vries, A. (2018). “Bitcoin’s growing energy problem”. In: *Joule* 2 (5), pp. 801–805. DOI: 10.1016/j.joule.2018.04.016.

de Vries, A. and C. Stoll (2021). “Bitcoin’s growing e-waste problem”. In: *Resources, Conservation and Recycling* 175, p. 105901. DOI: 10.1016/j.resconrec.2021.105901.

Dembo, A., S. Kannan, E. N. Tas, D. Tse, P. Viswanath, X. Wang, and O. Zeitouni (2020). “Everything is a race and Nakamoto always wins”. In: *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 859–878. DOI: 10.1145/3372297.3417290.

Deuber, D., B. Magri, and S. A. K. Thyagarajan (2019). "Redactable blockchain in the permissionless setting". In: *Symposium on Security and Privacy*. IEEE, pp. 124–138. DOI: 10.1109/sp.2019.00039.

Digiconomist (2022). *Bitcoin energy consumption index and Ethereum energy consumption index*. URL: https://digiconomist.net/.

Djamali, A., P. Dossow, M. Hinterstocker, B. Schellinger, J. Sedlmeir, F. Völter, and L. Willburger (2021). "Asset logging in the energy sector: A scalable blockchain-based data platform". In: *Energy Informatics* 4 (3). DOI: 10.1186/s42162-021-00183-3.

Drasch, B. J., G. Fridgen, T. Manner-Romberg, F. M. Nolting, and S. Radszuwill (2020). "The token's secret: the two-faced financial incentive of the token economy". In: *Electronic Markets* 30 (3), pp. 557–567. DOI: 10.1007/s12525-020-00412-9.

Drummer, D. and D. Neumann (2020). "Is code law? Current legal and technical adoption issues and remedies for blockchain-enabled smart contracts". In: *Journal of Information Technology* 35 (4), pp. 337–360. DOI: 10.1177/0268396220924669.

Du, W. D., S. L. Pan, D. E. Leidner, and W. Ying (2019). "Affordances, experimentation and actualization of FinTech: A blockchain implementation study". In: *The Journal of Strategic Information Systems* 28 (1), pp. 50–65. DOI: 10.1016/j.jsis.2018.10.002.

Dutta, P., T.-M. Choi, S. Somani, and R. Butala (2020). "Blockchain technology in supply chain operations: Applications, challenges and research opportunities". In: *Transportation Research Part E: Logistics and Transportation Review* 142, p. 102067. DOI: 10.1016/j.tre.2020.102067.

Egelund-Müller, B., M. Elsman, F. Henglein, and O. Ross (2017). "Automated execution of financial contracts on blockchains". In: *Business & Information Systems Engineering* 59 (6), pp. 457–467. DOI: 10.1007/s12599-017-0507-z.

Ether Alpha (2023). *Diversify now*. URL: https://clientdiversity.org/.

European Commission (2021). *European blockchain services infrastructure*. URL: https://digital-strategy.ec.europa.eu/en/policies/european-blockchain-services-infrastructure.

Eyal, I. and E. G. Sirer (2014). "Majority is not enough: Bitcoin mining is vulnerable". In: *International Conference on Financial Cryptography and Data Security*, pp. 436–454. DOI: 10.1007/978-3-662-45472-5_28.

Fang, M., X. Zhou, Z. Zhang, C. Jin, and A. Zhou (2022). "SEFrame: An SGX-enhanced smart contract execution framework for permissioned blockchain". In: *38th International Conference on Data Engineering*. IEEE, pp. 3166–3169. DOI: 10.1109/ICDE53745.2022.00289.

Feulner, S., T. Guggenberger, J.-C. Stoetzer, and N. Urbach (2022). “Shedding light on the blockchain disintermediation mystery: A review and future research agenda”. In: *Proceedings of the 26th European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2022_rp/13/.

Finck, M. (2018). “Blockchains and data protection in the European Union”. In: *European Data Protection Law Review* 4 (1), pp. 17–35. DOI: 10.21552/edpl/2018/1/6.

Fischer, M. J., N. A. Lynch, and M. S. Paterson (1985). “Impossibility of distributed consensus with one faulty process”. In: *Journal of the ACM* 32 (2), pp. 374–382. DOI: 10.1145/3149.214121.

Fridgen, G., N. Guggenberger, T. Hoeren, W. Prinz, N. Urbach, J. Baur, H. Brockmeyer, W. Gräther, E. Rabovskaja, V. Schlatt, A. Schweizer, J. Sedlmeir, and L. Wederhake (2019). *Opportunities and challenges of DLT (blockchain) in mobility and logistics*. URL: https://eref.uni-bayreuth.de/44302/.

Fridgen, G., F. Guggenmoos, J. Lockl, A. Rieger, and A. Schweizer (2018a). “Developing an evaluation framework for blockchain in the public sector: the example of the German asylum process”. In: *Proceedings of 1st ERCIM Blockchain Workshop 2018*. European Society for Socially Embedded Technologies. DOI: 10.18420/blockchain2018_10.

Fridgen, G., M.-F. Körner, S. Walters, and M. Weibelzahl (2021a). “Not all doom and gloom: How energy-intensive and temporally flexible data center applications may actually promote renewable energy sources”. In: *Business & Information Systems Engineering* 63 (3), pp. 243–256. DOI: 10.1007/s12599-021-00686-z.

Fridgen, G., R. Kräussl, O. Papageorgiou, and A. Tugnetti (2023). *The fundamental value of art NFTs*. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4337173.

Fridgen, G., J. Lockl, S. Radszuwill, A. Rieger, A. Schweizer, and N. Urbach (2018b). “A solution in search of a problem: A method for the development of blockchain use cases”. In: *Proceedings of the 24th Americas Conference on Information Systems*. AIS, pp. 3460–3469. URL: https://aisel.aisnet.org/amcis2018/StrategicIT/Presentations/14/.

Fridgen, G., S. Radszuwill, A. Schweizer, and N. Urbach (2021b). “Blockchain won’t kill the banks: Why disintermediation doesn’t work in international trade finance”. In: *Communications of the Association for Information Systems* 49, pp. 603–623. DOI: 10.17705/1cais.04932.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018c). “Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation". In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Fridgen, G., F. Regner, A. Schweizer, and N. Urbach (2018d). "Don't slip on the initial coin offering (ICO): A taxonomy for a blockchain-enabled form of crowdfunding". In: *Proceedings of the 26th European Conference on Information Systems*. AIS. URL: https://orbilu.uni.lu/handle/10993/44504.

Gallersdörfer, U., L. Klaaßen, and C. Stoll (2022). *Energy efficiency and carbon footprint of proof of stake blockchain protocols*. URL: https://www.carbon-ratings.com/dl/pos-report-2022.

Gallersdörfer, U., L. Klaaßen, and C. Stoll (2020). "Energy consumption of cryptocurrencies beyond Bitcoin". In: *Joule* 4 (9), pp. 1843–1846. DOI: 10.1016/j.joule.2020.07.013.

Garrido, G. M., J. Sedlmeir, Ö. Uludağ, I. S. Alaoui, A. Luckow, and F. Matthes (2022). "Revealing the landscape of privacy-enhancing technologies in the context of data markets for the IoT: A systematic literature review". In: *Journal of Network and Computer Applications* 207, p. 103465. DOI: 10.1016/j.jnca.2022.103465.

Gaži, P., A. Kiayias, and D. Zindros (2019). "Proof-of-stake sidechains". In: *Symposium on Security and Privacy*. IEEE, pp. 139–156. DOI: 10.1109/SP.2019.00040.

Georgiadis, E. (2019). *How many transactions per second can Bitcoin really handle? Theoretically.* URL: https://eprint.iacr.org/2019/416.

Gervais, A., G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun (2016). "On the security and performance of proof of work blockchains". In: *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security*, pp. 3–16. DOI: 10.1145/2976749.2978341.

Gluchowski, A. (2019). *Optimistic vs. ZK rollup: Deep dive*. URL: https://medium.com/matter-labs/optimistic-vs-zk-rollup-deep-dive-ea141e71e075.

Goldsby, C. and M. Hanisch (2022). "The boon and bane of blockchain: Getting the governance right". In: *California Management Review* 64 (3), pp. 141–168. DOI: 10.1177/00081256221080747.

Goodkind, A. L., B. A. Jones, and R. P. Berrens (2020). "Cryptodamages: Monetary value estimates of the air pollution and human health impacts of cryptocurrency mining". In: *Energy Research & Social Science* 59, p. 101281. DOI: 10.1016/j.erss.2019.101281.

Gorenflo, C., S. Lee, L. Golab, and S. Keshav (2020). "FastFabric: Scaling Hyperledger Fabric to 20 000 transactions per second". In: *International Journal of Network Management* 30 (5). DOI: 10.1002/nem.2099.

Gozman, D., J. Liebenau, and T. Aste (2020). "A case study of using blockchain technology in regulatory technology". In: *MIS Quarterly Executive* 19 (1), pp. 19–37. URL: https://aisel.aisnet.org/misqe/vol19/iss1/4.

Gramlich, V., T. Guggenberger, M. Principato, B. Schellinger, and N. Urbach (2023). "A multivocal literature review of decentralized finance". In: *Electronic Markets* 33. DOI: 10.1007/s12525-023-00637-4.

Gregor, S. and A. R. Hevner (2013). "Positioning and presenting design science research for maximum impact". In: *MIS Quarterly* 37 (2), pp. 337–355. DOI: 10.25300/misq/2013/37.2.01.

Groth, J. (2016). "On the size of pairing-based non-interactive arguments". In: *Advances in Cryptology – EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, pp. 305–326. DOI: 10.1007/978-3-662-49896-5_11.

Gudgeon, L., P. Moreno-Sanchez, S. Roos, P. McCorry, and A. Gervais (2020). "SoK: Layer-two blockchain protocols". In: *Financial Cryptography and Data Security: 24th International Conference*. Springer, pp. 201–226. DOI: 10.1007/978-3-030-51280-4_12.

Guggenberger, T., S. Hauffe, H. Huber, R. Ismer, Q. Jackl, A. Knipschild, D. Kühne, V. Schlatt, S. Schön, N. Urbach, and G. Wischrop (2020a). *SSI@LfSt: Einsatz der Blockchain-Technologie in der Steuerverwaltung*. URL: https://eref.uni-bayreuth.de/65188.

Guggenberger, T., L. Neubauer, J. Stramm, F. Völter, and T. Zwede (2023). "Accept me as I am or see me go: A qualitative analysis of user acceptance of self-sovereign identity applications". In: *Proceedings of the 56th Hawaii International Conference on System Sciences*, pp. 6560–6569. URL: https://hdl.handle.net/10125/103427.

Guggenberger, T., A. Schweizer, and N. Urbach (2020b). "Improving interorganizational information sharing for vendor managed inventory: Toward a decentralized information hub using blockchain technology". In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1074–1085. DOI: 10.1109/tem.2020.2978628.

Guggenberger, T., J.-C. Stoetzer, L. Theisinger, J. Amend, and N. Urbach (2021). "You can't manage what you can't define: The success of blockchain projects beyond the iron triangle". In: *Proccedings of the 42nd International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2021/fintech/fintech/3/.

Guggenmos, F., J. Lockl, A. Rieger, and G. Fridgen (2018). "Challenges and opportunities of blockchain-based platformization of digital identities in the public sector".

In: *Proceedings of the 26th European Conference on Information Systems*. AIS. URL: https://orbilu.uni.lu/handle/10993/44519.

Guggenmos, F., J. Lockl, A. Rieger, A. Wenninger, and G. Fridgen (2020). "How to develop a GDPR-compliant blockchain solution for cross-organizational workflow management: Evidence from the German asylum procedure". In: *Proceedings of the 53rd Hawaii International Conference on System Sciences*, pp. 4023–4032. DOI: 10.24251/hicss.2020.492.

Guo, Y.-M., Z.-L. Huang, J. Guo, X.-R. Guo, H. Li, M.-Y. Liu, S. Ezzeddine, and M. J. Nkeli (2021). "A bibliometric analysis and visualization of blockchain". In: *Future Generation Computer Systems* 116, pp. 316–332. DOI: 10.1016/j.future.2020.10.023.

Hacker, J., G. Miscione, T. Felder, and G. Schwabe (2023). "Commit or not? How blockchain consortia form and develop". In: *California Management Review*. DOI: 10.1177/00081256231175530.

Haldane, A. G. and R. M. May (2011). "Systemic risk in banking ecosystems". In: *Nature* 469 (7330), pp. 351–355. DOI: 10.1038/nature09659.

Haque, A. B., A. K. M. N. Islam, S. Hyrynsalmi, B. Naqvi, and K. Smolander (2021). "GDPR compliant blockchains – A systematic literature review". In: *IEEE Access* 9, pp. 50593–50606. DOI: 10.1109/access.2021.3069877.

Hayes, A. (2015). *A cost of production model for Bitcoin*. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2580904.

Hendershott, T., X. Zhang, J. L. Zhao, and Z. Zheng (2021). "FinTech as a game changer: Overview of research frontiers". In: *Information Systems Research* 32 (1). DOI: 10.1287/isre.2021.0997.

Hesse, M. and T. Teubner (2020). "Reputation portability–quo vadis?" In: *Electronic Markets* 30, pp. 331–349. DOI: 10.1007/s12525-019-00367-6.

Hevner, A., S. T. March, J. Park, S. Ram, et al. (2004). "Design science research in information systems". In: *MIS Quarterly* 28 (1), pp. 75–105. DOI: 10.2307/25148625.

Hoess, A., A. Rieger, T. Roth, G. Fridgen, and A. G. Young (2023). "Managing fashionable organizing visions: Evidence from the European Blockchain Services Infrastructure". In: *Proceedings of the 31st European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2023_rp/337/.

Hyvärinen, H., M. Risius, and G. Friis (2017). "A blockchain-based approach towards overcoming financial fraud in public sector services". In: *Business & Information Systems Engineering* 59 (6), pp. 441–456. DOI: 10.1007/s12599-017-0502-4.

Ilk, N., G. Shang, S. Fan, and J. L. Zhao (2021). “Stability of transaction fees in Bitcoin: A supply and demand perspective”. In: *MIS Quarterly* 45 (2), pp. 563–692. DOI: 10.25300/MISQ/2021/15718.

Jain, R. (1991). *The art of computer systems performance analysis – techniques for experimental design, measurement, simulation, and modeling*. Wiley.

Janssen, M., V. Weerakkody, E. Ismagilova, U. Sivarajah, and Z. Irani (2020). “A framework for analysing blockchain technology adoption: Integrating institutional, market and technical factors”. In: *International Journal of Information Management* 50, pp. 302–309. DOI: 10.1016/j.ijinfomgt.2019.08.012.

Javaid, M., A. Haleem, R. P. Singh, S. Khan, and R. Suman (2021). “Blockchain technology applications for Industry 4.0: A literature-based review”. In: *Blockchain: Research and Applications* 2 (4), p. 100027. DOI: 10.1016/j.bcra.2021.100027.

Jensen, T., J. Hedman, and S. Henningsson (2019). “How TradeLens delivers business value with blockchain technology”. In: *MIS Quarterly Executive* 18 (4), pp. 221–243. DOI: 10.17705/2msqe.00018.

Jeong, G., N. Lee, J. Kim, and H. Oh (2023). “Azeroth: Auditable zero-knowledge transactions in smart contracts”. In: *IEEE Access*. DOI: 10.1109/ACCESS.2023.3279408.

Jöhnk, J., T. Albrecht, L. Arnold, T. Guggenberger, L. Lämmermann, A. Schweizer, and N. Urbach (2021). “The rise of the machines: Conceptualizing the machine economy”. In: *Proceedings of the 25th Pacific Asia Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/pacis2021/54/.

Kamble, S. S., A. Gunasekaran, and R. Sharma (2020). “Modeling the blockchain enabled traceability in agriculture supply chain”. In: *International Journal of Information Management* 52, p. 101967. DOI: 10.1016/j.ijinfomgt.2019.05.023.

Kannengießer, N., S. Lins, T. Dehling, and A. Sunyaev (2020). “Trade-offs between distributed ledger technology characteristics”. In: *ACM Computing Surveys* 53 (2). DOI: 10.1145/3379463.

Kannengiesser, N., S. Lins, C. Sander, K. Winter, H. Frey, and A. Sunyaev (2021). “Challenges and common solutions in smart contract development”. In: *IEEE Transactions on Software Engineering* 48 (11), pp. 4291–4318. DOI: 10.1109/tse.2021.3116808.

Karger, E. (2020). “Combining blockchain and artificial intelligence – literature review and state of the art”. In: *Proceedings of the 41st International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2020/blockchain_fintech/blockchain_fintech/6/.

Kassen, M. (2021). “Understanding decentralized civic engagement: Focus on peer-to-peer and blockchain-driven perspectives on e-participation”. In: *Technology in Society* 66, p. 101650. DOI: 10.1016/j.techsoc.2021.101650.

Kiayias, A., A. Russell, B. David, and R. Oliynykov (2017). “Ouroboros: A provably secure proof-of-stake blockchain protocol”. In: *Advances in Cryptology -- CRYPTO 2017*, pp. 357–388. DOI: 10.1007/978-3-319-63688-7_12.

Köhler, S. and M. Pizzol (2020). “Technology assessment of blockchain-based technologies in the food supply chain”. In: *Journal of Cleaner Production* 269, p. 122193. DOI: 10.1016/j.jclepro.2020.122193.

Kolb, J., M. AbdelBaky, R. H. Katz, and D. E. Culler (2020). “Core concepts, challenges, and future directions in blockchain: A centralized tutorial”. In: *ACM Computing Surveys* 53 (1). DOI: 10.1145/3366370.

Kollmann, T., S. Hensellek, K. de Cruppe, and A. Sirges (2020). “Toward a renaissance of cooperatives fostered by blockchain on electronic marketplaces: A theory-driven case study approach”. In: *Electronic Markets* 30 (2), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

Körner, M.-F., J. Sedlmeir, M. Weibelzahl, G. Fridgen, M. Heine, and C. Neumann (2022). “Systemic risks in electricity systems: A perspective on the potential of digital technologies”. In: *Energy Policy* 164, p. 112901. DOI: 10.1016/j.enpol.2022.112901.

Kothapalli, A. and S. Setty (2023). *HyperNova: Recursive arguments for customizable constraint systems*. URL: https://eprint.iacr.org/2023/573.

Kranz, J., E. Nagel, and Y. Yoo (2019). “Blockchain token sale”. In: *Business & Information Systems Engineering* 61 (6), pp. 745–753. DOI: 10.1007/s12599-019-00598-z.

Krause, M. J. and T. Tolaymat (2018). “Quantification of energy and carbon costs for mining cryptocurrencies”. In: *Nature Sustainability* 1 (11), pp. 711–718. DOI: 10.1038/s41893-018-0152-7.

Lacity, M. C. (2018). “Addressing key challenges to making enterprise blockchain applications a reality”. In: *MIS Quarterly Executive* 17 (3), pp. 201–222. URL: https://aisel.aisnet.org/misqe/vol17/iss3/3.

Lamport, L., R. Shostak, and M. Pease (1982). “The Byzantine generals problem”. In: *ACM Transactions on Programming Languages and Systems* 4 (3), pp. 382–401. DOI: 10.1145/3335772.3335936.

Lavaur, T., J. Detchart, J. Lacan, and C. P. Chanel (2023). “Modular zk-rollup on-demand”. In: *Journal of Network and Computer Applications*, p. 103678. DOI: 10.1016/j.jnca.2023.103678.

Lee, S.-S., A. Murashkin, M. Derka, and J. Gorzny (2022). *SoK: Not quite water under the bridge: Review of cross-chain bridge hacks*. URL: https://eprint.iacr.org/2021/1589.

Lei, N., E. Masanet, and J. Koomey (2021). "Best practices for analyzing the direct energy use of blockchain technology systems: Review and policy recommendations". In: *Energy Policy* 156, p. 112422. DOI: 10.1016/j.enpol.2021.112422.

Leng, J., S. Ye, M. Zhou, J. L. Zhao, Q. Liu, W. Guo, W. Cao, and L. Fu (2020). "Blockchain-secured smart manufacturing in Industry 4.0: A survey". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 51 (1), pp. 237–252. DOI: 10.1109/tsmc.2020.3040789.

Leveling, J., M. Edelbrock, and B. Otto (2014). "Big data analytics for supply chain management". In: *International Conference on Industrial Engineering and Engineering Management*. IEEE, pp. 918–922. DOI: 10.1109/IEEM.2014.7058772.

Lichti, C. and A. Tumasjan (2023). ""My precious!": A values-affordances perspective on the adoption of Bitcoin". In: *Journal of the Association for Information Systems* 24 (3), pp. 629–663. DOI: 10.17705/1jais.00790.

Liu, C., H. Guo, M. Xu, S. Wang, D. Yu, J. Yu, and X. Cheng (2022). "Extending on-chain trust to off-chain–trustworthy blockchain data collection using trusted execution environment (TEE)". In: *IEEE Transactions on Computers* 71 (12), pp. 3268–3280. DOI: 10.1109/TC.2022.3148379.

Liu, Y., Q. Lu, L. Zhu, H.-Y. Paik, and M. Staples (2023). "A systematic literature review on blockchain governance". In: *Journal of Systems and Software* 197, p. 111576. DOI: 10.1016/j.jss.2022.111576.

Lockl, J., A. Rieger, G. Fridgen, M. Röglinger, and N. Urbach (2018). "Towards a theory of decentral digital process ecosystems: Evidence from the case of digital identities". In: *Workshop Blockchain Research? Beyond the Horizon*. URL: https://orbilu.uni.lu/handle/10993/44515.

Luu, L., J. Teutsch, R. Kulkarni, and P. Saxena (2015). "Demystifying incentives in the consensus computer". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 706–719. DOI: 10.1145/2810103.2813659.

March, S. T. and G. F. Smith (1995). "Design and natural science research on information technology". In: *Decision Support Systems* 15 (4), pp. 251–266. DOI: 10.1016/0167-9236(94)00041-2.

Marella, V., B. Upreti, J. Merikivi, and V. K. Tuunainen (2020). "Understanding the creation of trust in cryptocurrencies: The case of Bitcoin". In: *Electronic Markets* 30, pp. 259–271. DOI: 10.1007/s12525-019-00392-5.

Mattke, J., C. Maier, and A. Hund (2019). “How an enterprise blockchain application in the U.S. pharmaceuticals supply chain is saving lives”. In: *MIS Quarterly Executive* 18 (4), pp. 246–261. DOI: 10.17705/2msqe.00019.

Mazzoni, M., A. Corradi, and V. Di Nicola (2022). “Performance evaluation of permissioned blockchains for financial applications: The ConsenSys Quorum case study”. In: *Blockchain: Research and Applications* 3 (1), p. 100026. DOI: 10.1016/j.bcra.2021.100026.

McGhin, T., K.-K. R. Choo, C. Z. Liu, and D. He (2019). “Blockchain in healthcare applications: Research challenges and opportunities”. In: *Journal of Network and Computer Applications* 135, pp. 62–75. DOI: 10.1016/j.jnca.2019.02.027.

McMenamin, C. and V. Daza (2023). *Dynamic, private, anonymous, collateralizable commitments vs. MEV*. URL: https://arxiv.org/abs/2301.12818.

McMenamin, C., V. Daza, M. Fitzi, and P. O’Donoghue (2022). “Fairtradex: A decentralised exchange preventing value extraction”. In: *Proceedings of the ACM CCS Workshop on Decentralized Finance and Security*, pp. 39–46. DOI: 10.1145/3560832.3563439.

Monrat, A. A., O. Schelén, and K. Andersson (2019). “A survey of blockchain from the perspectives of applications, challenges, and opportunities”. In: *IEEE Access* 7, pp. 117134–117151. DOI: 10.1109/ACCESS.2019.2936094.

Mora, C., R. L. Rollins, K. Taladay, M. B. Kantar, M. K. Chock, M. Shimada, and E. C. Franklin (2018). “Bitcoin emissions alone could push global warming above 2°C”. In: *Nature Climate Change* 8 (11), pp. 931–933. DOI: 10.1038/s41558-018-0321-8.

Moyano, J. P. and O. Ross (2017). “KYC optimization using distributed ledger technology”. In: *Business & Information Systems Engineering* 59 (6), pp. 411–423. DOI: 10.1007/s12599-017-0504-2.

Mühle, A., A. Grüner, T. Gayvoronskaya, and C. Meinel (2018). “A survey on essential components of a self-sovereign identity”. In: *Computer Science Review* 30, pp. 80–86. DOI: 10.1016/j.cosrev.2018.10.002.

Muñoz, A., R. Ríos, R. Román, and J. López (2023). “A survey on the (in) security of trusted execution environments”. In: *Computers & Security* 129, p. 103180. DOI: 10.1016/j.cose.2023.103180.

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

Nofer, M., P. Gomber, O. Hinz, and D. Schiereck (2017). “Blockchain”. In: *Business & Information Systems Engineering* 59 (3), pp. 183–187. DOI: 10.1007/s12599-017-0467-3.

Notheisen, B., J. B. Cholewa, and A. P. Shanmugam (2017). “Trading real-world assets on blockchain”. In: *Business & Information Systems Engineering* 59 (6), pp. 425–440. DOI: 10.1007/s12599-017-0499-8.

O’Dwyer, K. J. and D. Malone (2014). “Bitcoin mining and its energy footprint”. In: *Proceedings of the 25th IET Irish Signals & Systems Conference 2014*, pp. 280–285. DOI: 10.1049/cp.2014.0699.

Ølnes, S. (2021). “Bitcoin and blockchain security – A study in misconceptions”. In: *Norwegian ICT Conference for Research and Education*. URL: https://ojs.bibsys.no/index.php/NIK/article/view/911/766.

Ølnes, S., J. Ubacht, and M. Janssen (2017). “Blockchain in government: Benefits and implications of distributed ledger technology for information sharing”. In: *Government Information Quarterly* 34 (3), pp. 355–364. DOI: 10.1016/j.giq.2017.09.007.

Ongaro, D. and J. Ousterhout (2014). “In search of an understandable consensus algorithm”. In: *USENIX Annual Technical Conference*, pp. 305–319. URL: https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf.

Ostern, N. K. (2019). “Blockchain in the IS research discipline: A discussion of terminology and concepts”. In: *Electronic Markets* 30, pp. 195–210. DOI: 10.1007/s12525-019-00387-2.

Otto, B. (2022). “A federated infrastructure for European data spaces”. In: *Communications of the ACM* 65 (4), pp. 44–45. DOI: 10.1145/3512341.

Ozdemir, A. and D. Boneh (2022). “Experimenting with collaborative zk-SNARKs: Zero-knowledge proofs for distributed secrets”. In: *31st USENIX Security Symposium*, pp. 4291–4308. DOI: https://www.usenix.org/system/files/usenixsecurity22-ozdemir.pdf.

Pedersen, A. B., M. Risius, and R. Beck (2019). “A ten-step decision path to determine when to use blockchain technologies”. In: *MIS Quarterly Executive* 18 (2), pp. 99–115. DOI: 10.17705/2msqe.0001.

Peffers, K., T. Tuunanen, M. A. Rothenberger, and S. Chatterjee (2007). “A design science research methodology for information systems research”. In: *Journal of Management Information Systems* 24 (3), pp. 45–77. DOI: 10.2753/mis0742-1222240302.

Perboli, G., V. Capocasale, and D. Gotta (2020). “Blockchain-based transaction management in Smart Logistics: A Sawtooth framework”. In: *Proceedings of the 44th Annual Computers, Software, and Applications Conference*. IEEE, pp. 1713–1718. DOI: 10.1109/compsac48688.2020.000-8.

Pereira, A. P. (2023). *Ethereum’s Beacon Chain updated after finality issues*. URL: https://cointelegraph.com/news/ethereum-s-beacon-chain-is-updated-after-finality-issues.

Pezzone, J. (2022). *Intel's SGX deprecation impacts DRM and Ultra HD Blu-ray support*. URL: https://www.techspot.com/news/93006-intel-sgx-deprecation-impacts-drm-ultra-hd-blu.html.

Platt, M., R. J. Bandara, A.-E. Drăgnoiu, and S. Krishnamoorthy (2021a). "Information privacy in decentralized applications". In: *Trust Models for Next-Generation Blockchain Ecosystems*. Springer, pp. 85–104. DOI: 10.1007/978-3-030-75107-4_4.

Platt, M. and P. McBurney (2021). "Sybil attacks on identity-augmented proof-of-stake". In: *Computer Networks* 199, p. 108424. DOI: 10.1016/j.comnet.2021.108424.

Platt, M. and P. McBurney (2023). "Sybil in the haystack: A comprehensive review of blockchain consensus mechanisms in search of strong Sybil attack resistance". In: *Algorithms* 16 (1), p. 34. DOI: 10.3390/a16010034.

Platt, M., J. Sedlmeir, D. Platt, J. Xu, P. Tasca, N. Vadgama, and J. I. Ibañez (2021b). "The energy footprint of blockchain consensus mechanisms beyond proof-of-work". In: *Companion Proceedings of the 21st International Conference on Software Quality, Reliability and Security*. IEEE, pp. 1135–1144. DOI: 10.1109/qrs-c55045.2021.00168.

Pocher, N., A. Vedder, C. Górriz López, and M. Palmirani (2023). *Distributed ledger technologies between anonymity and transparency: AML/CFT regulation of cryptocurrency ecosystems in the EU*.

Poon, J. and T. Dryja (2016). *The Bitcoin Lightning network: Scalable off-chain instant payments*. URL: https://www.bitcoinlightning.com/wp-content/uploads/2018/03/lightning-network-paper.pdf.

Putra, G. D., S. Malik, V. Dedeoglu, R. Jurda, and S. S. Kanhere (2023). "Challenges in designing blockchain for cyber-physical systems". In: *Communications of the ACM* 66 (7), pp. 81–82. DOI: 10.1145/3589648.

Qin, K., L. Zhou, B. Livshits, and A. Gervais (2021). "Attacking the DeFi ecosystem with flash loans for fun and profit". In: *Financial Cryptography and Data Security: 25th International Conference*. Springer, pp. 3–32. DOI: 10.1007/978-3-662-64322-8_1.

Regner, F., N. Urbach, and A. Schweizer (2019). "NFTs in practice – non-fungible tokens as core component of a blockchain-based event ticketing application". In: *Proceedings of the 39th International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2019/blockchain_fintech/blockchain_fintech/1/.

Renwick, R. and R. Gleasure (2020). "Those who control the code control the rules: How different perspectives of privacy are being written into the code of blockchain systems". In: *Journal of Information Technology* 36 (1), pp. 16–38. DOI: 10.1177/0268396220944406.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). "Building a blockchain application that complies with the EU general data protection regulation". In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Rieger, A., T. Roth, J. Sedlmeir, and G. Fridgen (2021). "The privacy challenge in the race for digital vaccination certificates". In: *Med* 2 (6), pp. 633–634. DOI: 10.1016/j.medj.2021.04.018.

Risius, M. and K. Spohrer (2017). "A blockchain research framework". In: *Business & Information Systems Engineering* 59 (6), pp. 385–409. DOI: 10.1007/s12599-017-0506-0.

Rossi, M., C. Mueller-Bloch, J. B. Thatcher, and R. Beck (2019). "Blockchain research in information systems: Current trends and an inclusive future research agenda". In: *Journal of the Association for Information Systems*, pp. 1388–1403. DOI: 10.17705/1jais.00571.

Roşu, I. and F. Saleh (2021). "Evolution of shares in a proof-of-stake cryptocurrency". In: *Management Science* 67 (2), pp. 661–672. DOI: 10.1287/mnsc.2020.3791.

Roth, T., A. Rieger, M. Utz, and A. G. Young (2022a). "The role of cultural fit in the adoption of fashionable IT: A blockchain case study". In: *Proceedings of the 43rd International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2022/blockchain/blockchain/17/.

Roth, T., A. Stohr, J. Amend, G. Fridgen, and A. Rieger (2023). "Blockchain as a driving force for federalism: A theory of cross-organizational task-technology fit". In: *International Journal of Information Management* 68, p. 102476. DOI: 10.1016/j.ijinfomgt.2022.102476.

Roth, T., M. Utz, F. Baumgarte, A. Rieger, J. Sedlmeir, and J. Strüker (2022b). "Electricity powered by blockchain: A review with a European perspective". In: *Applied Energy* 325, p. 119799. DOI: 10.1016/j.apenergy.2022.119799.

Roughgarden, T. (2021). "Transaction fee mechanism design". In: *ACM SIGecom Exchanges* 19 (1), pp. 52–55. DOI: 10.1145/3476436.3476445.

Ruan, P., T. T. Anh Dinh, Q. Lin, M. Zhang, G. Chen, and B. Chin Ooi (2020). "Revealing every story of data in blockchain systems". In: *ACM Sigmod Record* 49 (1), pp. 70–77. DOI: 10.1145/3422648.3422665.

Sai, A. R., J. Buckley, B. Fitzgerald, and A. Le Gear (2021). "Taxonomy of centralization in public blockchain systems: A systematic literature review". In: *Information Processing & Management* 58 (4), p. 102584. DOI: 10.1016/j.ipm.2021.102584.

Saleh, F. (2021). "Blockchain without waste: Proof-of-stake". In: *The Review of Financial Studies* 34 (3), pp. 1156–1190. DOI: 10.1093/rfs/hhaa075.

Sankagiri, S., X. Wang, S. Kannan, and P. Viswanath (2021). “Blockchain CAP theorem allows user-dependent adaptivity and finality”. In: *Financial Cryptography and Data Security: 25th International Conference*. Springer, pp. 84–103. DOI: 10.1007/978-3-662-64331-0_5.

Sartor, S., J. Sedlmeir, A. Rieger, and T. Roth (2022). “Love at first sight? A user experience study of self-sovereign identity wallets”. In: *Proceedings of the 30th European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2022_rp/46/.

Schellinger, B., F. Völter, J. Sedlmeir, and N. Urbach (2022). “Yes, I do: Marrying blockchain applications with GDPR”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 4631–4640. DOI: 10.24251/hicss.2022.563.

Schlatt, V., T. Guggenberger, J. Schmid, and N. Urbach (2022a). “Attacking the trust machine: Developing an information systems research agenda for blockchain cybersecurity”. In: *International Journal of Information Management*, p. 102470. DOI: 10.1016/j.ijinfomgt.2022.102470.

Schlatt, V., A. Schweizer, N. Urbach, and G. Fridgen (2016). *Blockchain: Grundlagen, Anwendungen und Potenziale*. URL: https://eref.uni-bayreuth.de/35561.

Schlatt, V., J. Sedlmeir, J. Traue, and F. Völter (2022b). “Harmonizing sensitive data exchange and double-spending prevention through blockchain and digital wallets: The case of e-prescription management”. In: *Distributed Ledger Technologies: Research and Practice* 2 (1). DOI: 10.1145/3571509.

Schneider, F. B. (1990). “Implementing fault-tolerant services using the state machine approach: A tutorial”. In: *ACM Computing Surveys* 22 (4), pp. 299–319. DOI: 10.1145/98163.98167.

Schütte, J., G. Fridgen, W. Prinz, T. Rose, N. Urbach, T. Hoeren, N. Guggenberger, C. Welzel, S. Holly, A. Schulte, P. Sprenger, C. Schwede, B. Weimert, B. Otto, M. Dalheimer, M. Wenzel, M. Kreutzer, M. Fritz, U. Leiner, and A. Nouak (2018). *Blockchain and smart contracts: Technologies, research issues and applications*. URL: https://eref.uni-bayreuth.de/45018.

Schwarz-Schilling, C., J. Neu, B. Monnot, A. Asgaonkar, E. N. Tas, and D. Tse (2022). “Three attacks on proof-of-stake Ethereum”. In: *Financial Cryptography and Data Security: 26th International Conference*. Springer, pp. 560–576. DOI: 10.1007/978-3-031-18283-9_28.

Schweizer, A., P. Knoll, N. Urbach, H. A. von der Gracht, and T. Hardjono (2020). “To what extent will blockchain drive the machine economy? Perspectives from a prospective study". In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1169–1183. DOI: 10.1109/tem.2020.2979286.

Schweizer, A., V. Schlatt, N. Urbach, and G. Fridgen (2017). "Unchaining social businesses-blockchain as the basic technology of a crowdlending platform". In: *Proceedings of the 37th International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2017/TransformingSociety/Presentations/8/.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020a). "Ein Blick auf aktuelle Entwicklungen bei Blockchains und deren Auswirkungen auf den Energieverbrauch". In: *Informatik Spektrum* 43 (6), pp. 391–404. DOI: 10.1007/s00287-020-01321-z.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020b). "The energy consumption of blockchain technology: Beyond myth". In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

Sedlmeir, J., R. Smethurst, A. Rieger, and G. Fridgen (2021). "Digital identities and verifiable credentials". In: *Business & Information Systems Engineering* 63 (5), pp. 603–613. DOI: 10.1007/s12599-021-00722-y.

Shehabi, A., S. Smith, D. Sartor, R. Brown, M. Herrlin, J. Koomey, E. Masanet, N. Horner, I. Azevedo, and W. Lintner (2016). *United States data center energy usage report*. Lawrence Berkeley National Laboratory. URL: https://eta-publications.lbl.gov/sites/default/files/lbnl-1005775_v2.pdf.

Shi, Z., H. Zhou, Y. Hu, S. Jayachander, C. de Laat, and Z. Zhao (2019). "Operating permissioned blockchain in clouds: A performance study of Hyperledger Sawtooth". In: *18th International Symposium on Parallel and Distributed Computing*. IEEE, pp. 50–57. DOI: 10.1109/ispdc.2019.00010.

Sislian, L. and A. Jaegler (2022). "Linkage of blockchain to enterprise resource planning systems for improving sustainable performance". In: *Business Strategy and the Environment* 31 (3), pp. 737–750. DOI: 10.1002/bse.2914.

Solana (2023). *Validator requirements*. URL: https://docs.solana.com/running-validator/validator-reqs.

Starknet (2021). *Redefining scalability*. URL: https://www.starknet.io/en/posts/engineering/redefining-scalability.

Stoll, C., L. Klaaßen, and U. Gallersdörfer (2019). "The carbon footprint of Bitcoin". In: *Joule* 3 (7), pp. 1647–1661. DOI: 10.1016/j.joule.2019.05.012.

Stradbrooke, S. (2023). *Tornado Cash coin mixer's governance lets its guard down, loses control*. URL: https://coingeek.com/tornado-cash-coin-mixer-governance-lets-its-guard-down-loses-control/.

Strüker, J., S. Albrecht, and S. Reichert (2019). “Blockchain in the energy sector”. In: *Business Transformation through Blockchain*. Springer, pp. 23–51. DOI: 10.1007/978-3-319-99058-3_2.

Sun Yin, H. H., K. Langenheldt, M. Harlev, R. R. Mukkamala, and R. Vatrapu (2019). “Regulating cryptocurrencies: A supervised machine learning approach to de-anonymizing the Bitcoin blockchain”. In: *Journal of Management Information Systems* 36 (1), pp. 37–73. DOI: 10.1080/07421222.2018.1550550.

Sunyaev, A., N. Kannengießer, R. Beck, H. Treiblmaier, M. Lacity, J. Kranz, G. Fridgen, U. Spankowski, and A. Luckow (2021). “Token economy”. In: *Business & Information Systems Engineering* 63 (4), pp. 457–478. DOI: 10.1007/s12599-021-00684-1.

Thakkar, P. and S. Natarajan (2021). “Scaling blockchains using pipelined execution and sparse peers”. In: *Proceedings of the Symposium on Cloud Computing*. ACM, pp. 489–502. DOI: 10.1145/3472883.3486975.

Thakkar, P., S. Nathan, and B. Viswanathan (2018). “Performance benchmarking and optimizing Hyperledger Fabric blockchain platform”. In: *26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, pp. 264–276. DOI: 10.1109/mascots.2018.00034.

Thaler, J. (2020). *Proofs, arguments, and zero-knowledge*. URL: https://people.cs.georgetown.edu/jthaler/ProofsArgsAndZK.pdf.

Thaler, J. (2022). *Measuring SNARK performance: Frontends, backends, and the future*. a16z Crypto. URL: https://a16zcrypto.com/measuring-snark-performance-frontends-backends-and-the-future/.

Thibault, L. T., T. Sarry, and A. S. Hafid (2022). “Blockchain scaling using rollups: A comprehensive survey”. In: *IEEE Access*. DOI: 10.1109/ACCESS.2022.3200051.

Toubiana, R., M. Macdonald, S. Rajananda, T. Lokvenec, T. C. Kingsley, and S. Romero-Brufau (2022). “Blockchain for electronic vaccine certificates: More cons than pros?” In: *Frontiers in Big Data* 5. DOI: 10.3389/fdata.2022.833196.

Toufaily, E., T. Zalan, and S. B. Dhaou (2021). “A framework of blockchain technology adoption: An investigation of challenges and expected value”. In: *Information & Management* 58 (3), p. 103444. DOI: 10.1016/j.im.2021.103444.

TradeLens Collaboration (2022). *Tradelens*. URL: https://www.tradelens.com/.

Truby, J. (2018). “Decarbonizing Bitcoin: Law and policy choices for reducing the energy consumption of blockchain technologies and digital currencies”. In: *Energy Research & Social Science* 44, pp. 399–410. DOI: 10.1016/j.erss.2018.06.009.

Truby, J., R. D. Brown, A. Dahdal, and I. Ibrahim (2022). “Blockchain, climate damage, and death: Policy interventions to reduce the carbon emissions, mortality, and net-zero implications of non-fungible tokens and Bitcoin”. In: *Energy Research & Social Science* 88, p. 102499.

Underwood, S. (2016). “Blockchain beyond Bitcoin”. In: *Communications of the ACM* 59 (11), pp. 15–17. DOI: 10.1145/2994581.

Upadhyay, N. (2020). “Demystifying blockchain: A critical analysis of challenges, applications, and opportunities”. In: *International Journal of Information Management* 54, p. 102120. DOI: 10.1016/j.ijinfomgt.2020.102120.

Utz, M., S. Albrecht, T. Zoerner, and J. Strüker (2018). “Blockchain-based management of shared energy assets using a smart contract ecosystem”. In: *Proceedings of the International Conference on Business Information Systems*. Springer, pp. 217–222. DOI: 10.1007/978-3-030-04849-5_19.

van Bokkem, D., R. Hageman, G. Koning, L. Nguyen, and N. Zarin (2019). *Self-sovereign identity solutions: The necessity of blockchain technology*. URL: https://arxiv.org/abs/1904.12816.

Verreydt, S., K. Yskout, and W. Joosen (2021). “Security and privacy requirements for electronic consent: A systematic literature review”. In: *ACM Transactions on Computing for Healthcare* 2 (2). DOI: 10.1145/3433995.

VISA (2019). *2019 corporate responsibility & sustainability report*. URL: https://usa.visa.com/dam/VCOM/download/corporate-responsibility/visa-2019-corporate-responsibility-report.pdf.

Vranken, H. (2017). “Sustainability of Bitcoin and blockchains”. In: *Current Opinion in Environmental Sustainability* 28. DOI: 10.1016/j.cosust.2017.04.011.

Vukolić, M. (2016). “The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication”. In: *Proceedings of the International Workshop on Open Problems in Network Security*. Springer, pp. 112–125. DOI: 10.1007/978-3-319-39028-4_9.

Wan, Z., X. Xia, D. Lo, J. Chen, X. Luo, and X. Yang (2021). “Smart contract security: A practitioners’ perspective”. In: *43rd International Conference on Software Engineering*. IEEE, pp. 1410–1422. DOI: 10.1109/ICSE43902.2021.00127.

Wang, C. and X. Chu (2020). “Performance characterization and bottleneck analysis of Hyperledger Fabric”. In: *Proceedings of the 40th International Conference on Distributed Computing Systems*. IEEE, pp. 1281–1286. DOI: 10.1109/icdcs47774.2020.00165.

Weigl, L., A. Amard, C. Codagnone, and G. Fridgen (2022a). “The EU’s digital identity policy: Tracing policy punctuations”. In: *Proceedings of the 15th International Conference on Theory and Practice of Electronic Governance*, pp. 74–81. DOI: 10.1145/3560107.3560121.

Weigl, L., T. Barbereau, J. Sedlmeir, and L. Zavolokina (2023). “Mediating the tension between data sharing and privacy: The case of DMA and GDPR”. In: *Proceedings of the 31st European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2023_rip/49/.

Weigl, L., T. J. Barbereau, A. Rieger, and G. Fridgen (2022b). “The social construction of self-sovereign identity: An extended model of interpretive flexibility”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 2543–2552. DOI: 10.24251/hicss.2022.316.

Weking, J., M. Mandalenakis, A. Hein, S. Hermes, M. Böhm, and H. Krcmar (2019). “The impact of blockchain technology on business models – a taxonomy and archetypal patterns”. In: *Electronic Markets* 30, pp. 285–305. DOI: 10.1007/s12525-019-00386-3.

Wüst, K. and A. Gervais (2018). “Do you need a blockchain?” In: *Crypto Valley Conference on Blockchain Technology*. IEEE. DOI: 10.1109/cvcbt.2018.00011.

Yeoh, P. (2017). “Regulatory issues in blockchain technology”. In: *Journal of Financial Regulation and Compliance* 25 (2), pp. 196–208. DOI: 10.1108/JFRC-08-2016-0068.

Yin, M., D. Malkhi, M. K. Reiter, G. G. Gueta, and I. Abraham (2019). “HotStuff: BFT consensus with linearity and responsiveness”. In: *Proceedings of the Symposium on Principles of Distributed Computing*. ACM, pp. 347–356. DOI: 10.1145/3293611.3331591.

Yu, M., S. Sahraei, S. Li, S. Avestimehr, S. Kannan, and P. Viswanath (2020). “Coded Merkle tree: Solving data availability attacks in blockchains”. In: *Financial Cryptography and Data Security: 24th International Conference*. Springer, pp. 114–134. DOI: 10.1007/978-3-030-51280-4_8.

Zamani, M., M. Movahedi, and M. Raykova (2018). “Rapidchain: Scaling blockchain via full sharding”. In: *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 931–948. DOI: 10.1145/3299869.3319889.

Zavolokina, L., G. Miscione, and G. Schwabe (2020a). “Buyers of ‘lemons’: How can a blockchain platform address buyers’ needs in the market for ‘lemons’?” In: *Electronic Markets* 30 (2), pp. 227–239. DOI: 10.1007/s12525-019-00380-9.

Zavolokina, L., R. Ziolkowski, I. Bauer, and G. Schwabe (2020b). “Management, governance and value creation in a blockchain consortium”. In: *MIS Quarterly Executive* 19 (1). DOI: 10.17705/2msqe.00022.

Zhang, R., R. Xue, and L. Liu (2019). “Security and privacy on blockchain”. In: *ACM Computing Surveys* 52 (3). DOI: 10.1145/3316481.

Zheng, Z., S. Xie, H. Dai, X. Chen, and H. Wang (2017). “An overview of blockchain technology: Architecture, consensus, and future trends”. In: *International Congress on Big Data*. IEEE, pp. 557–564. DOI: 10.1109/BigDataCongress.2017.85.

Ziolkowski, R., G. Miscione, and G. Schwabe (2020). “Decision problems in blockchain governance: Old wine in new bottles or walking in someone else’s shoes?” In: *Journal of Management Information Systems* 37 (2), pp. 316–348. DOI: 10.1080/07421222.2020.1759974.

A Overview of publications

A.1 Publications in this dissertation

A.1.1 List of publications

- **Research Paper 1**: J. Sedlmeir, H. U. Buhl, G. Fridgen, and R. Keller (2020). "The energy consumption of blockchain technology: Beyond myth". In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

  VHB-B, Scopus 91 % – Single first authorship. Most downloaded paper at BISE in 2021 and 2022.

- **Research Paper 2**: A. Rieger, T. Roth, J. Sedlmeir, and G. Fridgen (2022a). "We need a broader debate on the sustainability of blockchain". In: *Joule* 6 (6), pp. 1137–1141. DOI: 10.1016/j.joule.2022.04.013.

  Scopus: 99 % – Equal first authorship with Alexander Rieger and Tamara Roth.

- **Research Paper 3**: J. Sedlmeir, P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021a). "The DLPS: A new framework for benchmarking blockchains". In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

  VHB-C, CORE-A – Single first authorship. Best paper award in the Software & Technology track.

- **Research Paper 4**: T. Guggenberger, J. Sedlmeir, G. Fridgen, and A. Luckow (2022). "An in-depth investigation of the performance characteristics of Hyperledger Fabric". In: *Computers and Industrial Engineering* 173, p. 108716. DOI: 10.1016/j.cie.2022.108716.

  VHB-B, Scopus 95 % – Equal first authorship with Tobias Guggenberger.

- **Research Paper 5**: J. Sedlmeir, T. Wagner, E. Djerekarov, R. Green, J. Klepsch, and S. Rao (2022c). "A serverless distributed ledger for enterprises". In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 7382–7391. DOI: 10.24251/hicss.2022.886.

  VHB-C, CORE-A – Equal first authorship with Tim Wagner.

- **Research Paper 6**: J. Sedlmeir, J. Lautenschlager, G. Fridgen, and N. Urbach (2022b). "The transparency challenge of blockchain in organizations". In: *Electronic Markets* 32 (3), pp. 1779–1794. DOI: 10.1007/s12525-022-00536-0.

  VHB-B, Scopus 95 % – Equal first authorship with Jonathan Lautenschlager.

- **Research Paper 7**: J. Sedlmeir, T. Barbereau, J. Huber, L. Weigl, and T. Roth (2022a). "Transition pathways towards design principles of self-sovereign identity". In: *Proceedings of the 43rd International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2022/is_implement/is_implement/4/.

  VHB-A, CORE-A* – Single first authorship.

- **Research Paper 8**: V. Schlatt, J. Sedlmeir, S. Feulner, and N. Urbach (2022a). "Designing a framework for digital KYC processes built on blockchain-based self-sovereign identity". In: *Information & Management* 59 (7). DOI: 10.1016/j.im.2021.103553.

  VHB-B, Scopus 95 % – Equal first authorship with Vincent Schlatt.

A.1.2 Individual contributions

This dissertation is cumulative and comprises eight research papers. I wrote all of them together with several co-authors from multiple institutions and with different professional backgrounds. In this section, I will describe my individual contribution to each of the eight papers. The corresponding signed author statements are attached to the submission of this doctoral thesis. They also include additional details on the other authors' contributions.

In research paper 1 ("The energy consumption of blockchain technology: Beyond myth"), I was the single first author among a total of four co-authors. I conceptualized the overall structure and content of the paper and researched the core approaches toward modeling and estimating the energy consumption characteristics of different blockchains. This analysis involved a thorough literature review and comprehensive collection of empirical data on the corresponding technical and economic parameters, such as average block time, coin prices, and transaction fees. From these models and the collected data, I derived lower- and upper-bound estimates for different PoW cryptocurrencies according to the previously identified models. I also generated estimates for the energy consumption of PoS and permissioned blockchains as well as centralized databases from a first series of experiments with selected hardware. From all of this data, I conceptualized and created the two figures. I was responsible for writing the initial draft of the manuscript and led the iterative refinement for the initial submission and during the revision process. All co-authors provided academic guidance, gave critical feedback, and edited the manuscript.

In research paper 2 ("We need a broader debate on the sustainability of blockchain"), I was one of the three first authors among a total of four authors. All authors contributed to the conceptualization of the manuscript. The two other first authors were responsible for the initial draft of a major share of the manuscript. I was responsible for designing the experiments for measuring the energy consumption of the corresponding permissioned blockchains, allocating the funding for the required cloud services, and collecting and evaluating data through a literature study and experiments with the DLPS (which I developed for research paper 3). The data collection also involved researching the electricity consumption characteristics of different servers. I created the corresponding figures and wrote the first draft for the supplemental information section that details how the energy consumption figures were obtained. All authors contributed equally to the iterative refinement of the manuscript and the supplemental information.

In research paper 3 ("The DLPS: A new framework for benchmarking blockchains"), all six authors equally conceptualized the research project. I took the role of the single first author of this paper. I conceptualized generic benchmarking capabilities, in particular an algorithm to automatically determine maximum throughput, latency, and resource utilization metrics, and implemented this algorithm as a major generic component of the benchmarking framework. The second author implemented the initial AWS integration to facilitate the automatic startup of variable numbers of servers that after bootstrapping become blockchain nodes or clients. The second author also supported the initial tests of the benchmarking framework and implemented startup scripts for private Ethereum networks. I implemented startup scripts for the four other blockchain networks (Hyperledger Fabric, Hyperledger Indy, Quorum, and Hyperledger Sawtooth) as well as the corresponding smart contracts and client networks. I then collected all data required for the paper in iterative refinements of the benchmarking framework and via conducting and evaluating preliminary and final experiments. Based on the data analysis, I created the figures and wrote the major share of the initial draft of the manuscript. The fourth author contributed to the initial version of the introduction and background. All authors provided critical feedback and contributed to the iterative refinement of the manuscript before the initial submission and during the revision process.

In research paper 4 ("An in-depth investigation of the performance characteristics of Hyperledger Fabric"), I was one of the two first authors among a total of four authors. All authors conceptualized the paper. The other first author was mainly responsible for writing the introduction, background, and related work sections. He also conducted the systematic literature review that builds the basis for the related work section. Together with the other first author, I extended the DLPS developed in research paper 3 to account for additional features of Hyperledger Fabric, such as private collections. I incorporated further blockchain-agnostic features into the DLPS, such as intercontinental deployments and robustness tests. I performed all experiments, evaluated the collected data, created the corresponding charts, and wrote the research method and results section. All authors validated the experimental results. I created the initial version of the discussion and conclusion sections with the other first author. All authors provided critical feedback and contributed equally to the revisions of the manuscript.

In research paper 5 ("A serverless distributed ledger for enterprises"), there were six authors in total, with me among the two first authors. All authors contributed to the initial idea of writing a paper about the serverless blockchain product that the other first author and the sixth author had developed commercially. The second first author and I conceptualized the paper, supported by the third author. The other first and sixth author led the requirements analysis by interviewing representatives from different organizations. I conducted the literature review, analyzed related work, and performed the qualitative and quantitative criteria-based evaluation sections. The data collection for the quantitative evaluation involved integrating the serverless blockchain implementation into the DLPS, which I did with the support of the other first author and the third author. The other first author and I created the initial draft of the paper. The other first author and the fourth author created the architecture-related figure. I conducted the performance analysis and evaluated the corresponding data together with the other first author and the fourth author. All authors contributed equally to the iterative refinement of the manuscript.

In research paper 6 (“The transparency challenge of blockchain in organizations”), I was one of the two first authors. All authors contributed to the conceptualization of the paper. I reviewed the literature on data protection in the context of blockchain technology, structured the different aspects of the transparency challenge of blockchain systems used in organizations, and explored the properties and business implications of potential cryptographic solutions to these problems. The other first author systematically collected information about various blockchain projects in practice, categorized them into application patterns, and surveyed examples for the corresponding sensitive data blockchain-based information systems need to process. The third and fourth authors contributed to the application patterns for blockchain projects in enterprises. The other first author created an initial draft for the introduction and background sections. I wrote the main sections that describe the privacy challenge and solution approaches with different cryptographic techniques, as well as the conclusion section. All authors provided critical feedback and contributed to the iterative refinement of the manuscript. I guided the two revision rounds and implemented the major share of the reviewers’ suggestions.

In research paper 7 (“Transition pathways towards design principles of self-sovereign identity”), I was the single first author and responsible for the research project’s conceptualization and execution. I guided and supported the second author with the systematic literature review and in the data collection and design iteration phases. I also provided the technical background on the different flavors of decentralized identity management and created the figure on historical developments related to SSI. The third and fourth authors led the development of transition pathways in science and technology studies (STS) as the theoretical lens underlying the paper. The fifth author refined the DSR method and the analysis of the corresponding interview data collected by me and the second author. The fifth author was also responsible for drafting the results and discussion sections. All authors provided critical feedback and contributed to the iterative refinement of the manuscript.

In research paper 8 (“Designing a framework for digital KYC processes built on blockchain-based self-sovereign identity”), there were four authors. The first author and I contributed equally. The other first author and the third author conceptualized the research project. The third author conducted most of the interviews and surveyed related work. The other first author and I supported and extended the initial interview-based data collection and analysis. The other first author contributed the major part of the method section. I contributed to the technical background and analyzed, improved, and described the technical properties of the artifact. I also derived the design principles from an analysis of the interview transcripts. Together with the other first author, I created the initial manuscript draft and refined the design objectives, method, criteria-based evaluation, and design principles sections. All authors provided critical feedback and contributed equally to the iterative refinement of the manuscript.

A.2 Other publications not included in this dissertation

Over the course of the dissertation, I also co-authored the following journal and conference publications, book chapters, and industry reports. These papers are not part of this doctoral thesis.

A.2.1 Journal publications

- J. Glöckler, J. Sedlmeir, M. Frank, and G. Fridgen (2023). *How self-sovereign identity can improve enterprises identity and access management*. Accepted at *Business & Information Systems Engineering*. (VHB-B, Scopus 91 %)

- A. Hoess, J. Lautenschlager, J. Sedlmeir, G. Fridgen, V. Schlatt, and N. Urbach (2023). *Toward seamless mobility-as-a-service: Providing multimodal mobility through digital wallets*. Accepted with minor revisions at *Business & Information Systems Engineering*. (VHB-B, Scopus 91 %)

- E. Hartwich, A. Rieger, J. Sedlmeir, D. Jurek, and G. Fridgen (2023). “Machine economies”. In: *Electronic Markets* 33. DOI: 10.1007/s12525-023-00649-0. (VHB-B, Scopus 95 %)

- T. Barbereau, R. Smethurst, O. Papageorgiou, J. Sedlmeir, and G. Fridgen (2023b). “Decentralised finance’s timocratic governance: The distribution and exercise of tokenised voting rights". In: *Technology in Society* 73, p. 102251. DOI: 10.1016/j.techsoc.2023.102251. (Scopus 98 %)

- V. Schlatt, J. Sedlmeir, J. Traue, and F. Völter (2022b). "Harmonizing sensitive data exchange and double-spending prevention through blockchain and digital wallets: The case of e-prescription management". In: *Distributed Ledger Technologies: Research and Practice* 2 (1). DOI: 10.1145/3571509.

- T. Roth, M. Utz, F. Baumgarte, A. Rieger, J. Sedlmeir, and J. Strüker (2022). "Electricity powered by blockchain: A review with a European perspective". In: *Applied Energy* 325, p. 119799. DOI: 10.1016/j.apenergy.2022.119799. (Scopus 99 %)

- M. Babel, V. Gramlich, M.-F. Körner, J. Sedlmeir, J. Strüker, and T. Zwede (2022). "Enabling end-to-end digital carbon emission tracing with shielded NFTs". In: *Energy Informatics* 5 (1). DOI: 10.1186/s42162-022-00199-3. (Scopus 56 %)

- S. Feulner, J. Sedlmeir, V. Schlatt, and N. Urbach (2022). "Exploring the use of self-sovereign identity for event ticketing systems". In: *Electronic Markets* 32 (3), pp. 1759–1777. DOI: 10.1007/s12525-022-00573-9. (VHB-B, Scopus 95 %)

- M.-F. Körner, J. Sedlmeir, M. Weibelzahl, G. Fridgen, M. Heine, and C. Neumann (2022). "Systemic risks in electricity systems: A perspective on the potential of digital technologies". In: *Energy Policy* 164, p. 112901. DOI: 10.1016/j.enpol.2022.112901. (VHB-B, Scopus 97 %)

- J. Sedlmeir, F. Völter, and J. Strüker (2021b). "The next stage of green electricity labeling: Using zero-knowledge proofs for blockchain-based certificates of origin and use". In: *ACM SIGENERGY Energy Informatics Review* 1 (1), pp. 20–31. DOI: 10.1145/3508467.3508470.

- G. M. Garrido, J. Sedlmeir, Ö. Uludağ, I. S. Alaoui, A. Luckow, and F. Matthes (2022). "Revealing the landscape of privacy-enhancing technologies in the context of data markets for the IoT: A systematic literature review". In: *Journal of Network and Computer Applications* 207, p. 103465. DOI: 10.1016/j.jnca.2022.103465. (Scopus 98 %)

- T. Rückel, J. Sedlmeir, and P. Hofmann (2022). "Fairness, integrity, and privacy in a scalable blockchain-based federated learning system". In: *Computer Networks* 202, p. 108621. DOI: 10.1016/j.comnet.2021.108621. (Scopus 92 %)

- A. Djamali, P. Dossow, M. Hinterstocker, B. Schellinger, J. Sedlmeir, F. Völter, and L. Willburger (2021). “Asset logging in the energy sector: A scalable blockchain-based data platform”. In: *Energy Informatics* 4 (3). DOI: 10.1186/s42162-021-00183-3. (Scopus 56 %)

- T. Guggenberger, J. Lockl, M. Röglinger, V. Schlatt, J. Sedlmeir, J.-C. Stoetzer, N. Urbach, and F. Völter (2021). “Emerging digital technologies to combat future crises: learnings from COVID-19 to be prepared for the future”. In: *International Journal of Innovation and Technology Management* 18 (4), p. 2140002. DOI: 10.1142/S0219877021400022 (Scopus 50 %)

- P. Schott, J. Sedlmeir, N. Strobel, T. Weber, G. Fridgen, and E. Abele (2019). “A generic data model for describing flexibility in power markets”. In: *Energies* 12 (10), p. 1893. DOI: 10.3390/en12101893. (Scopus 83 %)

A.2.2 Conference publications

- L. Weigl, T. Barbereau, J. Sedlmeir, and L. Zavolokina (2023). “Mediating the tension between data sharing and privacy: The case of DMA and GDPR”. in: *Proceedings of the 31st European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2023_rip/49/. (VHB-B, CORE-A)

- E. Ermolaev, I. Abellán Álvarez, J. Sedlmeir, and G. Fridgen (2023). “z-Commerce: Designing a data-minimizing one-click checkout solution”. In: *18th International Conference on Design Science Research in Information Systems and Technology*. Springer. DOI: 10.1007/978-3-031-32808-4_1. (VHB-C, CORE-A)

- J. Sedlmeir, T. Barbereau, J. Huber, L. Weigl, and T. Roth (2022a). “Transition pathways towards design principles of self-sovereign identity”. In: *Proceedings of the 43rd International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2022/is_implement/is_implement/4/. (VHB-A, CORE-A*)

- G. Munilla-Garrido, J. Sedlmeir, and M. Babel (2022). “Towards verifiable differentially-private polling”. In: *Proceedings of the International Conference on Availability, Reliability and Security*. ACM. DOI: 10.1145/3538969.3538992. (CORE-B)

- S. Sartor, J. Sedlmeir, A. Rieger, and T. Roth (2022). “Love at first sight? A user experience study of self-sovereign identity wallets”. In: *Proceedings of the 30th*

European Conference on Information Systems. AIS. URL: https://aisel.aisnet.org/ecis2022_rp/46/. (VHB-B, CORE-A)

- A. Hoess, T. Roth, J. Sedlmeir, G. Fridgen, and A. Rieger (2022). “With or without blockchain? Towards a decentralized, SSI-based eRoaming architecture”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 4621–4630. DOI: 10.24251/hicss.2022.562. (VHB-C, CORE-A)

- B. Schellinger, F. Völter, J. Sedlmeir, and N. Urbach (2022b). “Yes, I do: Marrying blockchain applications with GDPR”. in: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 4631–4640. DOI: 10.24251/hicss.2022.563. (VHB-C, CORE-A)

- M. Platt, J. Sedlmeir, D. Platt, J. Xu, P. Tasca, N. Vadgama, and J. I. Ibañez (2021). “The energy footprint of blockchain consensus mechanisms beyond proof-of-work”. In: *Companion Proceedings of the 21st International Conference on Software Quality, Reliability and Security*. IEEE, pp. 1135–1144. DOI: 10.1109/qrs-c55045.2021.00168. (CORE-B)

- P. Seitz, E. Abele, L. Bank, T. Bauernhansl, E. Colangelo, G. Fridgen, J. Schilp, P. Schott, J. Sedlmeir, N. Strobel, and T. Weber (2019). “IT-based architecture for power market oriented optimization at multiple levels in production processes”. In: *Procedia CIRP*. vol. 81. Elsevier, pp. 618–623. DOI: 10.1016/j.procir.2019.03.165.

A.2.3 Research commentaries

- A. Rieger, T. Roth, J. Sedlmeir, L. Weigl, and G. Fridgen (2022b). “Not yet another digital identity”. In: *Nature Human Behaviour* 6 (1), pp. 3–3. DOI: 10.1038/s41562-021-01243-0. (Scopus 99 %)

- A. Rieger, T. Roth, J. Sedlmeir, and G. Fridgen (2021). “The privacy challenge in the race for digital vaccination certificates”. In: *Med* 2 (6), pp. 633–634. DOI: 10.1016/j.medj.2021.04.018. (Scopus 95 %)

- J. Sedlmeir (2020). “Von Bitcoin zu Libra und dem digitalen Euro: Technische Fortschritte von Blockchains und deren Implikationen auf digitale Währungen”. In: *Recht der Zahlungsdienste* 3, pp. 210–213.

A.2.4 Book chapters

- T. Barbereau, J. Sedlmeir, R. Smethurst, G. Fridgen, and A. Rieger (2022). "Tokenization and regulatory compliance for art and collectibles markets". In: *Blockchains and the Token Economy: Studies in Theory and Practice*. Ed. by M. Lacity and H. Treiblmaier. Palgrave Macmillan. Chap. 8. DOI: 10.1007/978-3-030-95108-5_8.

- J. Baur, J. Brügmann, J. Sedlmeir, and N. Urbach (2021). "Kryptowährungen – Grundlegende technische, wirtschaftliche und rechtliche Aspekte". In: *Handbuch IT Recht*. Vol. 4, pp. 1551–1577.

- T. Bauernhansl, D. Bauer, E. Abele, R. Ahrens, L. Bank, M. Brugger, E. Colangelo, H. Eigenbrod, G. Fridgen, F. G. Vasquez, A. Grigorjan, M. Jarke, R. Keller, R. Lodwig, J. Pullmann, G. Reinhart, M. Rösch, A. Sauer, D. Schel, A. Schlereth, P. Schott, F. Schulz, J. Sedlmeir, P. Seitz, P. Simon, and T. Weber (2019). "Graphitherstellung". In: *Energieflexibilität in der deutschen Industrie: Ergebnisse aus dem Kopernikus-Projekt-Synchronisierte und energieadaptive Produktionstechnik zur flexiblen Ausrichtung von Industrieprozessen auf eine fluktuierende Energieversorgung (SynErgie)*. Ed. by A. Sauer, E. Abele, and H. U. Buhl. Fraunhofer Verlag, pp. 245–313.

A.2.5 Industry reports

- V. Gramlich, M. Principato, B. Schellinger, J. Sedlmeir, J. Amend, J. Stramm, T. Zwede, J. Strüker, and N. Urbach (2022). *Decentralized finance (DeFi): Foundations, applications, potentials, and challenges*. URL: https://eref.uni-bayreuth.de/70709.

- B. Schellinger, J. Sedlmeir, L. Willburger, J. Strüker, and N. Urbach (2022a). *Mythbusting Self-Sovereign Identity (SSI): Diskussionspapier zu selbstbestimmten digitalen Identitäten*. URL: https://eref.uni-bayreuth.de/68552.

- C. Gola and J. Sedlmeir (2022). *Addressing the sustainability of distributed ledger technology*. Questioni di Economia e Finanza (Occasional Papers). Bank of Italy. URL: https://www.bancaditalia.it/pubblicazioni/qef/2022-0670/.

- J. Strüker, M. Utz, and J. Sedlmeir (2022). *Einsatz der Blockchain-Technologie für Smart Grid Dienstleistungen durch E-PKW im Reallabor „EnStadt:Pfaff“.*

- J. Strüker, N. Urbach, T. Guggenberger, J. Lautenschlager, N. Ruhland, V. Schlatt, J. Sedlmeir, J.-C. Stoetzer, and F. Völter (2021). *Self-Sovereign Identity : Grundlagen, Anwendungen und Potenziale portabler digitaler Identitäten*. URL: https://eref.uni-bayreuth.de/66090.

- J. Amend, M. Federbusch, G. Fridgen, F. Köhler, A. Rieger, V. Schlatt, J. Sedlmeir, A. Stohr, and C. van Dun (2021). *Digitization of certification processes in the asylum procedure by means of digital identities: A feasibility study by Germany's Federal Office for Migration and Refugees*. URL: https://orbilu.uni.lu/bitstream/10993/48332/1/blockchain-whitepaper-2021.pdf.

- J. Sedlmeir (2021a). *Supply chain reference implementation – use case parts traceability*. Full report available for MOBI members. URL: https://dlt.mobi/wp-content/uploads/2021/06/MOBI-SC0004RI2021-Version-1.0-compressed-LINKED.pdf.

- J. Sedlmeir (2021b). *Vehicle identity II reference implementation architecture – use cases vehicle registration and maintenance*. Full report available for MOBI members. URL: https://dlt.mobi/wp-content/uploads/2021/02/MOBI-VID0004-Reference-Implementation-Architecture-2021.pdf.

- G. Fridgen, N. Guggenberger, T. Hoeren, W. Prinz, N. Urbach, J. Baur, H. Brockmeyer, W. Gräther, E. Rabovskaja, V. Schlatt, A. Schweizer, J. Sedlmeir, and L. Wederhake (2019). *Opportunities and challenges of DLT (blockchain) in mobility and logistics*. URL: https://eref.uni-bayreuth.de/44302/.

A.2.6 Conditionally accepted publications and preprints

- T. Barbereau, E. Ermolaev, M. Brennecke, E. Hartwich, and J. Sedlmeir (2023a). *Beyond a fistful of tumblers: Toward a multi-layered taxonomy of Ethereum-based crypto-asset mixers* – conditional accept at *International Conference on Information Systems*. (VHB-A, CORE-A*)

- S. Napirata, J. Sedlmeir, A. Rieger, G. Fridgen, and S. Zimmermann (2023). *Competition between centralized and decentralized platforms: The case of seamless mobility* – conditional accept at *International Conference on Information Systems*. (VHB-A, CORE-A*)

- M. Babel and J. Sedlmeir (2023). *Bringing data minimization to digital wallets at scale with general-purpose zero-knowledge proofs*. URL: https://arxiv.org/abs/2301.00823
- M. Platt, S. Ojeka, A.-E. Drăgnoiu, O. E. Ibelegbu, F. Pierangeli, and J. Sedlmeir (2022). *How to make users adopt more sustainable cryptocurrencies: Evidence from Nigeria*. URL: https://arxiv.org/abs/2208.00280
- J. Groß, J. Sedlmeir, M. Babel, A. Bechtel, and B. Schellinger (2021). *Designing a central bank digital currency with support for cash-like privacy*. URL: https://papers.ssrn.com/abstract=3891121

B Research Paper 1 – The energy consumption of blockchain technology: Beyond myth

Authors:

Johannes Sedlmeir, Hans Ulrich Buhl, Robert Keller, & Gilbert Fridgen

Published in:

Business & Information Systems Engineering (DOI: 10.1007/s12599-020-00656-x)

Abstract:

When talking about blockchain technology in academia, business, and society, one still frequently hears generalizations made about its – supposedly inherent – enormous energy consumption. This perception inevitably raises concerns about the further adoption of blockchain technology, which inhibits rapid uptake of what is widely considered to be a groundbreaking and disruptive innovation. However, blockchain technology is far from homogeneous, meaning that blanket statements about its energy consumption should be reviewed with care. This article is meant to bring clarity to the topic in a holistic fashion, looking beyond claims about the energy consumption of Bitcoin, which have, so far, dominated the discussion.

Keywords:

Blockchain, cryptocurrency, energy consumption, distributed ledger technology, sustainability

B.1 Introduction

Blockchain technology entered public awareness with its first application, the cryptocurrency Bitcoin (Nakamoto, 2008), which was established in 2009 and currently exhibits a market capitalization of more than 100 billion USD. In the last decade, Blockchain technology has developed significantly and is now implemented in a wide range of scenarios, including Ethereum or Hyperledger Fabric, which allow distributed platforms to function with unprecedented versatility (Lockl et al., 2020). Consequently, many researchers and practitioners have realized that blockchain technology holds disruptive potential beyond its use in cryptocurrencies (Beck, 2018; Fridgen et al., 2018a; Labazova et al., 2019). Generally speaking, blockchain technology permits secure transactions to be made without the involvement of intermediaries, and is, therefore, appealing to individuals as well as to industry and the public sector. However, Bitcoin still dominates many people's perceptions of blockchain technology. Moreover, it is well-known that Bitcoin consumes a massive amount of energy[1] (de Vries, 2018). Consequently, one frequently encounters claims that the energy consumption of blockchain technology in general is problematic (Truby, 2018). Considering the current discussions regarding climate change and sustainability, these statements could therefore inhibit or delay widespread adoption of blockchain technology (Beck et al., 2018).

This article challenges the common prejudices regarding the energy consumption of the supposedly homogeneous blockchain technology by providing a detailed analysis of current scientific knowledge. It, thereby, addresses the energy consumption of IS, in general a subject for which BISE traditionally takes responsibility (Buhl and Jetter, 2009; Schmidt et al., 2009). In particular, it also addresses the need for a detailed investigation of the energy consumption of blockchain technology, as pointed out in Beck et al. (2017). In Section B.2, we first provide some technical background on proof-of-work (PoW) blockchains and determine the level of their energy consumption. Using these estimates, we illustrate that today's PoW cryptocurrencies do, indeed, consume an amount of energy which may be regarded as disproportionate when compared to the currencies' actual utility. However, we also argue that the energy consumption associated with widespread uptake of PoW cryptocurrencies is not likely to become a major threat to the climate in the future. In Section B.3, we put these results into perspective by presenting blockchains with alternative consensus mechanisms. We illustrate that these kinds

[1] Strictly speaking, we cannot consume energy, but merely change its form from valuable (e.g., electricity) to less valuable (e.g., heat) energy. Nevertheless, we will stick to the common usage of the phrase here.

of blockchain technology already consume several orders of magnitude less energy than the first generation PoW blockchains and that these blockchains, thus, largely mitigate the energy problem. However, we argue that, in addition to consensus, the redundancy underlying all types of blockchain technology can make blockchain-based IT solutions considerably more energy-intensive than a non-blockchain, centralized alternative. In Section B.4, we discuss this issue and also give an overview of methods and concepts which could further decrease the energy consumption of blockchain technology. In Section B.5, we illustrate our findings by a first rough comparison of the energy consumption of some non-blockchain, centralized systems to that of basic blockchain architectures. We conclude with with an outlook and suggested topics for further research in Section B.6.

B.2 Proof-of-work blockchains

B.2.1 Technological basics

Bitcoin, the first application built on blockchain technology, is a decentralized payment system in which all participating computers ("nodes") store a copy – or, more precisely, a replica, since there is no distinguished master – of the associated ledger. A ledger is commonly defined as a collection of accounts, stating one's current rights of ownership of a particular asset – in the case of Bitcoin, units of the eponymous cryptocurrency. The underlying technology, blockchain, provides a means to store information chronologically and redundantly on a decentralized database, and an agreement process through which the nodes synchronize and modify their global state ("operate transactions") (Crosby et al., 2016). It is, therefore, not exclusively suitable for use with cryptocurrencies, but can be applied to many processes in which the involvement of an intermediary such as a bank, a notary, or any (digital) platform owner is not desirable.

Blockchains, in general, achieve this synchronization by linking transactions to form batches ("blocks") and adding these, sequentially, to the existing linear data structure ("chain"). Utilizing Merkle trees and hash-pointers, this data structure is highly tamper-sensitive, making retrospective manipulations easy to detect. Agreement about which new blocks to append is reached using a so-called consensus mechanism. Anyone can run a node for the common cryptocurrencies and participate in the consensus mechanism of their underlying blockchains using public key cryptography and hence without any form of registration. Consequently, blockchains underlying such open systems, which allow for unrestricted access and participation, are termed *permissionless*. Since, on a permissionless blockchain, the inclusion of a distinct entity to provide accounts and passwords is not viable, authentication based on a public key infrastructure is highly suitable. For such blockchains, a simple voting-based agreement process based on “one man – one vote” is not secure, since a potential attacker could simply create multiple accounts to gain a majority and take control of the system; this is called a Sybil attack (Douceur, 2002).

Bitcoin’s key innovation was to provide a suitable consensus mechanism for use in this scenario. Specifically, Bitcoin combined several well-known concepts from cryptography to form the so-called PoW. This refers to the right to create a new block from a subset of queued transactions when one finds a solution to a cryptographic, computationally intensive puzzle. The process of searching for a solution is called “mining”. This achieves the coupling of voting weight to a scarce resource – computing power and thus energy – and hence prevents Sybil attacks. The mining process is economically incentivized in that participants are rewarded for every valid block that is found and disseminated. The reward typically consists of a certain amount of the associated cryptocurrency and the fees for the associated transactions. The value of the former is proportional to the cryptocurrency’s market price, so the success of cryptocurrencies on financial markets in the last years has provided a very strong incentive to participate in mining. In turn, this has led to enormous energy consumption associated with the underlying PoW blockchains.

It is essential to note that the high energy consumption of PoW blockchains is the result of neither inefficient algorithms nor outdated hardware. Strikingly, such blockchains are “energy-intensive by design”. It is their high energy consumption that protects PoW blockchains from attacks: Depending on the scenario, an attacker must bear at least 25 to 50% of the total computing power that participating miners use for mining – and, thus, the same proportion of the total energy consumption (under the assumption of equal hardware) – to be able to successfully manipulate or control the system (Eyal and Sirer, 2014). Consequently, the more valuable a PoW cryptocurrency is, the better it is protected against attacks, confirming that PoW is, indeed, a thoughtful design.

B.2.2 General estimates

Starting with the work of O’Dwyer and Malone (2014), researchers have analyzed the energy consumption caused by Bitcoin in numerous scientific publications over recent years (Stoll et al., 2019). However, results on the energy consumption of PoW cryptocurrencies and blockchain technology in general are rare. Determining the exact value for the energy consumption of a multitude of open, distributed networks is a hard task because the precise number of participants, the properties of their hardware, and the effort which they put into mining are unknown. Fortunately, however, one can obtain good estimates for a lower and an upper bound of the energy consumption of any PoW blockchain by following Vranken (2017) and Krause and Tolaymat (2018): Since both the difficulty of the cryptographic puzzles and the frequency at which solutions are found are easily observable, one can calculate the expected value of the minimum frequency of calculations ("hash-rate") needed to solve the puzzles as often as observed. This gives a lower bound of the energy consumption of an arbitrary PoW blockchain:

$$\text{total power consumption} \geq \text{total hash rate} \times \text{min energy per hash.} \tag{1}$$

This estimate indicates the lower bound, reflecting the likelihood that more solutions are found than disseminated, that further computations – in addition to mining – are being carried out, and that not every miner has the most energy-efficient hardware.

Both the current hash rate of a public blockchain and the energy efficiency of the most efficient mining hardware can easily be retrieved from online material. However, one must be aware that mining hardware is in general blockchain-dependent because the algorithms used for hashing can differ. For example, Bitcoin uses SHA256, for which very efficient application-specific integrated circuits (ASICS) exist, i.e., chips that are highly optimized for computing hash values and, thus, for solving the puzzles. On the other hand, Ethereum was designed to prevent the use of highly specific mining hardware, so general-purpose GPUs can be used for mining. Note that (1) does not depend on any other parameters and, therefore, gives a very reliable lower bound. Entering the current numbers – retrieved from Coinmarketcap (2020) and Coinswitch (2019) on 2020-02-05 – into (1) yields a lower bound for power consumption of 6.8 GW, which equates to an annual energy requirement of at least 60 TWh. Alternatively, one could, of course, also integrate the time-dependent lower bound over the period under consideration.

One can also determine an upper bound for the energy requirement of the mining process for a PoW blockchain, assuming honest and rational miners whose utility from mining is solely financial profit: Participation in the mining process is only profitable as long as the expected revenue from mining is higher than the associated costs:

$$\begin{aligned} \text{mining rewards} + \text{transaction fees} &= \text{tot. mining revenue} \\ &\geq \text{tot. mining costs} \\ &\geq \text{tot. energy consumption} \times \text{min. electricity price.} \end{aligned}$$

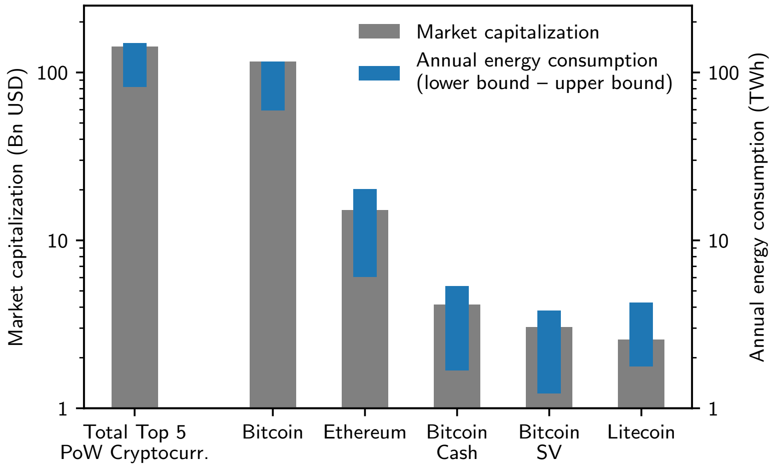


Figure 1: Market capitalization and the computed bounds on energy consumption for the 5 highest valued Proof-of-Work cryptocurrencies. Note the logarithmic scale on the y-axis.

A few easy manipulations yield the desired upper bound:

$$\text{total power consumption} \leq \frac{\text{block reward} \times \text{coin price} + \text{transaction fees}}{\text{avg. blocktime} \times \text{min. electricity price}}. \tag{2}$$

As hardware costs represent a substantial part of the costs side, and electricity prices vary significantly around the globe, we cannot assume that the upper bound is very tight. The block reward, i.e., the number of cryptocurrency coins one receives for solving a puzzle, the price of a coin, and current transaction fees are, again, publicly observable for every PoW cryptocurrency, meaning that only sensitive number which has to be estimated is the minimum electricity price. de Vries (2018), for example, argues that $0.05\,\frac{\text{USD}}{\text{kWh}}$ is a reasonable lower bound for electricity prices. This gives an upper bound of approximately 125 TWh per year for the energy consumption of Bitcoin, using data from Coinmarketcap (2020) for 2020-02-05.

We repeated the calculation of the lower bound (1) and the upper bound (2) for the remaining 4 PoW cryptocurrencies with market capitalization of at least 1 billion USD. Figure 1 displays the resultant ranges for their respective energy consumption. We see that the lower and upper bounds are, in general, quite close and, therefore, represent a meaningful estimate of the actual energy consumption for each of the 5 major PoW cryptocurrencies. A manifestation of this fact could be observed when in the course of a general drop in financial markets due to the Corona pandemic, market prices for Bitcoin dropped by up to 40 % in March 2020. This implies a drop of the upper bound (2) in our model by the same rate, and, indeed, the total hash rate was observed to drop by approximately 30 % shortly after: Seemingly, mining was no longer profitable for some miners at this point (Beincrypto, 2020). This incident also illustrates that the upper bound is highly sensitive on the economic circumstances: Assuming that electricity prices dropped by the same rate as the prices for cryptocurrencies – which is in fact conceivable in an economic crisis – the upper bound (2) would remain unchanged. On the other hand, if electricity prices generally dropped by 50 %, e.g., due to decreased demand or increased feed-in of renewables, or a rush for cryptocurrencies led to an increase of their prices by 100 % and, therefore, to a level that we have already observed by the beginning of 2018, our upper bound would double in each of the scenarios, and even quadruple if both happened to occur at the same time. Consequently, we learn that we cannot take for granted that the given upper bound holds forever; it merely represents a snapshot for the current economic situation.

We also observe that the expected energy consumption of the 5 investigated cryptocurrencies strongly correlates with their market capitalization, which makes sense since parameters, such as block reward per time are comparable among the cryptocurrencies and total transaction fees are generally low compared to block rewards.

Moreover, the total market capitalization for all other PoW cryptocurrencies is significantly lower than that of Bitcoin itself. This indicates that the total energy consumption of *all* PoW cryptocurrencies other than Bitcoin will fall below our upper bound for the energy consumption of Bitcoin. A more precise estimate could be obtained by applying (2) to all remaining PoW cryptocurrencies. This would, however, be a tedious task, as one would have to collect specific parameters, such as block reward and average block time, for each PoW cryptocurrency, of which there are currently more than 1000.

In both estimates, we have, so far, only taken into account the energy consumption involved in mining, i.e., solving the cryptographic puzzles, and neglected the energy consumption of the other tasks which have to be performed on the participating nodes, mainly, validating new blocks and updating their local databases accordingly. This is, in fact, a reasonable approximation: for the lower bound, we only lose some tightness. To justify the validity of our upper bound, we argue that the energy consumption associated with maintaining the nodes, mining excluded, is, in fact, negligible compared to the energy consumption of mining for today's major PoW blockchains: To validate a single block in today's cryptocurrencies, every node must typically download up to a few Megabytes of data and perform as many as several thousand hash computations, as well as a comparable number of corresponding computations and database operations. For example, in a 1 MB block used in Bitcoin, there can only be a maximum of around 2000 transactions. These are the leaves of the Merkle tree and, therefore, give a total of 4000 hash value computations and a similar number of corresponding database manipulations. By comparison, finding a single block currently involves around 10^{23} hash computations to solve a puzzle in Bitcoin, around 10^{20} hash computations for Bitcoin Cash and Bitcoin SV, and around 10^{15} hash computations for Ethereum and Litecoin. Even for a million nodes – and taking into account differences in efficiency between common and specialized mining hardware, given that ASICS can be millions of times more efficient than CPUs at computing hashes – the energy consumption associated with mining is still *orders of magnitude* higher than the energy consumption required to maintain the nodes (de Vries, 2018).

At this point, it is important to emphasize that further increasing the energy efficiency of mining hardware would not reduce a PoW blockchain's energy requirements in the long term: To keep the average time for solving a puzzle constant, and, hence, to ensure the security and constant functionality of the network, the difficulty of the cryptographic puzzles is periodically adapted to the total computing power of the network. Since energy costs outweigh hardware costs in the long run, participants with improved hardware can solve more puzzles at the same energy costs. Other participants have to follow suit with the competition. This, in turn, involves higher overall computing power, and means that the difficulty of the puzzle needs to be increased so that it is, on average, solved as frequently as before. Hence, it is only in the (short-term) conversion phase that positive effects are conceivable. In fact, competition in the mining hardware market, resulting from the hype around cryptocurrencies, has dramatically increased the energy efficiency of mining hardware in the last decade. In the long term, it is to be expected that even with groundbreaking innovation in the energy efficiency of mining hardware, Bitcoin's and other PoW blockchains' energy requirements will remain at the previous level unless the remaining economic quantities on the right-hand side of (2) change considerably.

B.2.3 Closing notes on the energy consumption of PoW blockchains

In summary, our lower and upper bounds represent different approaches and use different quantities that have to be estimated. Yet, these bounds are very consistent in the case of all of the cryptocurrencies we investigated. For example, we determined electricity consumption to be between 60 and 125 TWh per year for Bitcoin. This is in the range of the annual electricity consumption of countries such as Austria (75 TWh) and Norway (125 TWh). However, as cryptocurrencies currently process only few transactions per second, the theoretical limit is typically in the low two- or three-digit range, e.g., approx. 15 for Ethereum and Bitcoin and 100 for Bitcoin Cash. This is primarily determined by the parameters 'average block time', 'minimum size of transactions', and

’maximum block size’ (Georgiadis, 2019). Accordingly, a single transaction currently requires enough electrical energy to meet the needs of the average size German household for weeks, or even months. By contrast, traditional payment systems process, on average, thousands of transactions per second, and as many as tens of thousands at peak times. In their publication in “Nature Climate Change”, Mora et al. (2018) extrapolate the energy consumption of a single Bitcoin transaction to the order of magnitude required for handling payments on a global scale. They claim that if Bitcoin were to handle the number of transactions required by a worldwide payment system, the associated emissions alone would lead to a global temperature increase of 2°C in the coming decades. However – as has already been pointed out in a critical ’Matters Arising’ response by Dittmar and Praktiknjo (2019) – when increasing the blocksize and, therefore, the throughput, according to our previous arguments, the energy consumption associated with mining would remain constant, and the energy consumption associated with the remaining tasks would still be negligible. This means that, overall, there would be no noticeable increase in total energy consumption. This argument is, however, based on the assumption that the economic quantities from the estimate of the upper bound (2), namely, the prices for electricity and the respective cryptocurrency, remain constant.

In practice, however, the blocks cannot be enlarged at will. While in Bitcoin Cash, for example, the blocksize has been increased by a factor of 8 (compared to Bitcoin) without any problems, a significantly larger block size is currently not practicable. This is because, the larger a block is, the longer it takes for it to be propagated by the worldwide blockchain network. This can have a negative effect for latency (the time it takes to distribute a new block to all nodes) and, also, security: More solutions to the puzzles are likely to be found as a certain block propagates through the network, splitting the honest miners’ resources and, therefore, leaving the network more vulnerable to attack. Moreover, not every household can afford a high bandwidth and large hardware storage, so higher requirements can also lead to a lower degree of decentralization. This trade-off has already been discussed, e.g., in Bitcoin Magazine (2018). If, however, storage capacities (hard disks) and network speed continue to improve worldwide, a considerable increase in block sizes might be conceivable in the future. This would enable higher transaction rates without a noticeable increase in energy consumption.

Finally, for most PoW blockchains, the block reward is not constant, but periodically halved, typically, every few years. Since mining fees are currently negligible compared to block rewards, the upper bound (2) is proportional to the electricity price and block reward. Hence, if the prices for crypto-coins and electricity prices remain at the same level, one could even expect that in the long run, the energy consumption of PoW blockchains will also halve in each of these periods, until the rewards from mining are comparable to the total transaction fees.

We conclude that, although the energy consumption of PoW blockchains is arguably enormous in relation to their technical performance, it does not represent an essential threat to the climate, even if significantly more transactions are processed in the future. Moreover, since the area of application of most blockchains – and, in particular, the major cryptocurrencies – is often far beyond payments, plenty of opportunities for new ecosystems and business models arise. An evaluation should therefore not only compare performance metrics and energy consumption, but also take into account the unique opportunities offered by this technology.

B.3 Alternative consensus mechanisms

Fortunately, the PoW consensus mechanism, which – as already described – was designed to be energy-intensive, is not the only way to achieve consensus in a distributed system. The probably best-known alternative for the permissionless systems required for cryptocurrencies and other open decentralized applications is the so-called proof-of-stake (PoS) consensus mechanism. In this case, the weight of a participant's vote is not tied to the scarce resource of computing power, but to the scarce resource of capital (see Section B.2.1 on why coupling with a scarce resource is necessary). More precisely, there is a random mechanism[2] that determines who is allowed to build ("mint", "forge", "bake") and attach the next block. With the help of this mechanism, the probability of being selected is linked to the amount of cryptocurrency that the node has deposited and locked ("staked") for this purpose. The deposit also incentivizes the node to stick to the rules of the network, as any misbehaviour detected will lead to the node losing this deposit. The advantage of PoS is that it does not involve any computationally intensive steps such as solving the cryptographic puzzles in PoW. The computational complexity of PoS consensus is low and, typically, insensitive to network size. It is, therefore, very energy-efficient for large-scale systems. Accordingly, based on our arguments regarding the energy consumption associated with operating transactions in Section B.2, the energy consumption of PoS blockchains is several orders of magnitude lower than that of

[2] There are no truly random number generators for classical computers, but, as a first approximation, this heuristics provides a good indication. The pseudo-randomness typically comes from a subset of the previous blocks.

PoW. It is primarily for this reason that the community of the cryptocurrency with the currently second-highest market capitalization, Ethereum, is trying to switch from PoW to PoS. Other cryptocurrencies, such as EOS, Tezos, and TRON – all of which feature in the Top 20 cryptocurrencies in terms of market capitalization – are already successfully using PoS. There are, however, controversial discussions in the community. Some argue that getting rid of PoW's energy consumption comes at the price of security, e.g., because one can only accrue voting weight (capital) from inside the system. However, one can also argue that PoS has less of a tendency to centralize (mining has economies of scale) and is, thus, more secure in the long run. We will not enter in this discussion up here but want to highlight that the outcome will likely decide which consensus-type for permissionless blockchains prevails and, therefore, impacts the energy consumption of future open decentralized applications.

On the other hand, blockchain technology can also be useful in constellations in which only a restricted group of participants take part in consensus. These are referred to as *permissioned* blockchains. They are of particular interest to many industries and, also, to the public sector: participants usually build a consortium, and there is a registration process meaning that all of the participants in consensus are known (Fridgen et al., 2018b; Rieger et al., 2019). Therefore, it is not necessary to tie voting weight to a scarce resource here, and one can reach a consensus using some kind of election in which everyone has a single vote. Therefore, this kind of consensus mechanism is sometimes called proof-of-identity or, very often, proof-of-authority (PoA). The term PoA usually involves different levels of security, from mathematically proven and long-established, fully fault-tolerant mechanisms (Paxos, PBFT) over heuristically-secure algorithms, such as Istanbul BFT and Aura, to basic crash-tolerant mechanisms such as RAFT (Angelis et al., 2017). Popular implementations of such permissioned blockchains are Hyperledger Fabric and Quorum. The more secure these PoA consensus mechanisms are, the greater their complexity and, therefore, the greater their energy consumption. For example, PBFT consensus overhead scales at least quadratically with respect to the number of nodes in the network and is hence – by contrast to PoW and PoS – highly sensitive on the network size. This, in turn, correlates with the energy consumption associated with consensus.

Beyond these popular consensus mechanisms, there are several more, an overview of which is provided by Eklund and Beck (2019). An example is proof-of-elapsed-time, which intends to establish trusted random number generators through secure hardware modules. As PoS and PoA, these further concepts typically do not involve a cryptographic puzzle, except for some concepts which try to establish some kind of "useful proof-of-work" which solves puzzles that are in some way meaningful for business or science. Since many of these types of consensus mechanisms are not currently prevalent in relevant applications, and because they usually have low energy requirements compared to PoW, we will not investigate these consensus mechanisms in more detail.

The main result of the discussion about blockchains with alternative consensus mechanisms is that, by getting rid of energy intensity by design, their energy consumption is orders of magnitude lower compared to PoW-blockchains. Consequently, the energy consumption of non-PoW blockchains can hardly be considered problematic for the climate. Yet, beyond PoW and, thus, on a completely different scale, the type of consensus mechanism can have a significant impact on energy consumption.

B.4 The impact of redundancy on energy consumption

We have already seen that a portion of blockchains' energy consumption relates to consensus, and another portion relates to redundant operations. We have seen that for PoW blockchains, the energy consumption related to consensus outweighs the energy consumption associated with operating transactions, so the redundancy aspect is usually not discussed in detail. For non-PoW blockchains, however, the energy consumption related to consensus is no more enormous, and, therefore, the contribution to total energy consumption by redundant operations may be significant. Hence, it is not only alternative consensus mechanisms that one should look at to further reduce the energy consumption of blockchain technology, but also concepts which allow reduced operation redundancy. Generally speaking, the primary motivations behind all of the concepts presented in this section that may help to reduce redundancy are increased scalability, throughput, and privacy for blockchain solutions. Conveniently, these all happen to reduce the degree of redundancy and, therefore, improve the overall energy consumption.

We can distinguish between two approaches to reducing redundancy: reducing the *degree* of redundancy, i.e., the number of nodes that perform certain operations, and the *workload* associated with operating a transaction. In attempts to reduce the degree of redundancy, a concept called *sharding* is often mentioned. Sharding is about splitting the nodes in the network into subsets ("shards") and processing each transaction on only one of these subsets. How easily sharding can be achieved largely depends on the consensus mechanism. For example, sharding is very difficult to apply to PoW blockchains, because one has to make sure that, within a shard, computing power is roughly equally distributed to maintain a balance of voting weight among the associated nodes. In a PoS blockchain, voting power is tied to the capital deposited by each node. This information is publicly available and can, therefore, be freely used in creation of shards. Other concepts to reduce the degree of redundancy include off-chain payment channels between two parties who repeatedly interact. Such channels usually require a transaction on the Blockchain, in the course of which off-chain payment channels are created and terminated. Ideally, however, all interim transactions are operated purely bilateral and do not involve a transaction on the corresponding blockchain. That is to say that, ideally, only balances, or accumulated deltas signed by the members on the payment hub, are periodically recorded on-chain. Payment hubs, a generalization of payment channels to multiple parties, e.g., Nocust, or connections between them, e.g., Lightning for Bitcoin or Raiden for Ethereum, are the focus of active research (Gudgeon et al., 2020). A similar basic concept is the use of sidechains (e.g., Plasma for Ethereum). These are small blockchain networks which periodically refer to the main chain as a highly reliable root. Generally speaking, however, reducing the degree of redundancy also makes a blockchain network more centralized and must, therefore, be carefully weighed against concerns about security, liveness, and trust. Finding a good compromise between these interests could enable a reduction of the total workload in the system, and, therefore, a reduction of its total energy consumption.

On the other hand, the *workload* associated with redundant operations, e.g., the verification of new blocks, can be significantly reduced, which also mitigates the redundancy issue. One very straightforward improvement is, therefore, optimization of the computational complexity of the used cryptographic algorithms, e.g., for verifying signatures. Yet, this has some natural limits: Currently, transactions are operated “naively” on all nodes in the sense that all transaction-related data must be provided on-chain and all nodes recompute every step on their own. This could be significantly improved by storing and verifying only short correctness proofs on a blockchain and distributing the larger, plaintext data on another layer to the relevant participants. In particular, SNARKS, STARKS, and other (zero-knowledge) proofs of computational integrity which require much less verification and communication overhead on-chain seem very promising (Ben-Sasson et al., 2019). This is because, unlike methods that lower the degree of redundancy – these do likely not have a negative impact on security because every transaction is still verified by every node.

In summary, there are various ways to reduce the intrinsic redundancy of blockchains and, therefore, to reduce also their energy consumption. The relative energy saving potential is, however, negligible for PoW blockchains as the energy consumption of mining dominates all other contributions. However, it may still be relatively high for networks in which consensus is not energy-intensive, in particular, if the network is large.

B.5 A first comparison of different architectures

We can now use our results from the previous chapters to make a first comparison of the energy consumption of typical blockchain architectures. The role of consensus has already been discussed in Section B.3, where we suggested that a major distinction should be made between PoW and non-PoW blockchains, although the differences between other consensus mechanisms might also be significant. On the other hand, for small networks, redundancy does not add much absolute energy consumption, particularly when compared to the scale of PoW blockchains' energy consumption. By contrast, for large systems consisting of many nodes, the natural redundancy in a blockchain can lead to much higher energy consumption. If a PoS or alternative non-PoW blockchain replaces Bitcoin or another PoW cryptocurrency in the future, we have to expect that there will still be tens of thousands of nodes. Although the energy consumption of such a network will be negligible compared to Bitcoin, it will, therefore, remain high compared to a non-blockchain centralized system with minimal redundancy (i.e., because of backups). Figure 2 illustrates this observation and gives a rough comparison of the energy consumption of different architectures, using selected centralized systems as a baseline. We decided to display the energy per transaction. However, as discussed in Section B.2, this is not an ideal metric for PoW blockchains but does correctly represent the order of magnitude.

We arrived at our estimates in the following way: A simple key-value store such as LevelDB can sustainably operate tens of thousands of transactions per second on office hardware with a power consumption of less than 100 W (own measurements), which yields less than 10^{-2} J per transaction. A more complex database, such as CouchDB, with one backup still manages more than 10^{3} transactions per second on the same hardware, resulting in at most 0.1 J per transaction (own measurements). As an example of a small-scale enterprise blockchain, we refer to a Hyperledger Fabric architecture with 10 nodes, each on cloud instances with 32 vCPUs and therefore likely consuming a few thousand Watts in total. According to Androulaki et al. (2018), such a system can handle around 3000 transactions per second, so we arrive at an order of magnitude of 1 J per transaction. On the other hand, an Ethereum full node on Geth which does not mine consumes approximately 0.1 J for a simple payment transaction, depending on whether or not idle power consumption is taken into account (own measurements). This seems low, but

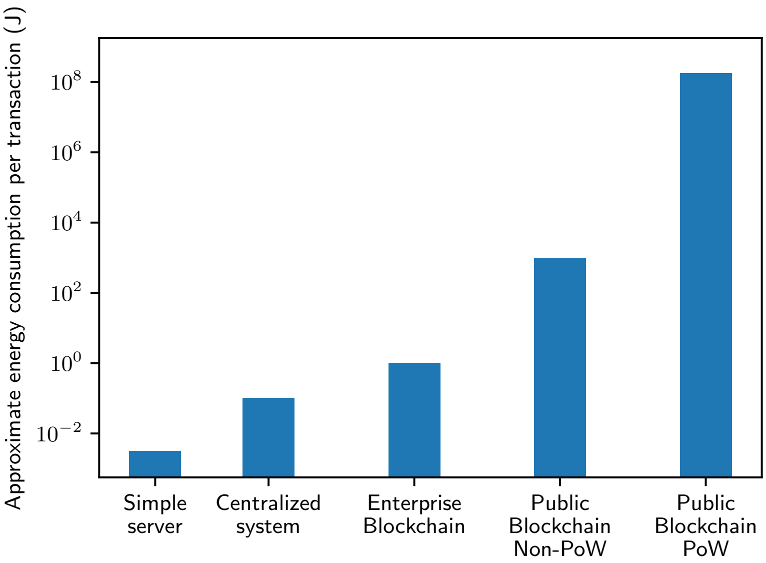


Figure 2: A rough comparison of the order of magnitude of energy consumption per transaction for different architectures. A simple server can operate transactions with very low energy consumption. A typical non-blockchain, centralized system in applications will use a more complex database and backups, thus mildly increasing the energy consumption. A small-scale permissioned blockchain as used in cross-enterprise use-cases has a similar degree of redundancy, but some additional yet limited overhead due to, e.g., PoA consensus and more complex cryptographic operations. A non-PoW permissionless blockchain with a large number of nodes can already exhibit a significantly increased energy consumption due to the high degree of redundancy. However, compared to a major PoW blockchain, energy consumption is still negligible.

in a network of 10^4 nodes, which is approximately the number of active full nodes in Bitcoin or Ethereum, this amounts to approximately 10^3 J per transaction, which is already orders of magnitude more than for the described centralized systems and small-scale enterprise blockchain. However, it is still many orders of magnitude less than for the current PoW blockchains such as Bitcoin with about 10^9 J per transaction. All numbers given here should be taken with caution as they are highly dependent on the precise architecture, security measures, type of hardware, and other parameters. They should therefore be regarded a ballpark estimate, and reliable numbers have yet to be established. We suggest this interesting topic for further work, including a more thorough investigation of the role of consensus mechanism and the energy efficiency of transactions depending on transaction type or choice of blockchain implementation. For permissioned blockchains, this might be particularly relevant when enterprises have to decide for or against a particular blockchain implementation.

B.6 Conclusion

In this article, we first analyzed the energy consumption of today's prevailing PoW blockchains, which underly most cryptocurrencies. While their energy consumption is, indeed, massive, particularly when compared to the number of transactions they can operate, we found that they do not pose a large threat to the climate, mainly because the energy consumption of PoW blockchains does not increase substantially when they process more transactions. We also argued that although the energy consumption of non-PoW blockchains and in particular permissioned blockchains which are used in enterprise context is generally considerably higher than that of non-blockchain, centralized systems, it is many orders of magnitude lower than that of PoW cryptocurrencies such as Bitcoin. We also observed a close interrelationship between security aspects and the choice of consensus mechanism and redundancy characteristics, and therefore, energy consumption. Hence, we conclude that further investigation in this direction, which has many similarities to Vitalik Buterin's "scalability trilemma", might help to find the best compromise between performance, security, and energy consumption.

Our contribution demonstrates that the energy consumption of blockchain technology differs significantly between different design choices. Consequently, it is an important dimension to consider during the conception of a blockchain-based IT solution (Kannengießer et al., 2019). We argued that using blockchain technology with non-PoW consensus – which is the case in an increasing number of business applications – already substantially mitigates sustainability issues. However, we also illustrated that due to consensus and inherent redundancy, blockchain-based solutions in general still require more energy than non-blockchain, centralized architectures. However, in enterprise applications, blockchains are typically only one part of a hybrid solution in which most processes are operated via conventional IT, and little information which is relevant to the remaining participants on the blockchain is processed on-chain (Rieger et al., 2019). Reducing the workflows operated on-chain to a minimum, therefore, also mitigates concerns about the energy consumption. On the other hand, we know from other areas of IT that significant energy savings can be enabled by process optimization and digitization. As there are plenty of scenarios in which blockchain technology might finally turn out to be an enabler of further digitization of processes, the increase in energy consumption of a specific blockchain must always be weighed against the savings it provides. For example, by enabling the digitization of supply-chain processes, blockchain can substantially reduce the amount of paperwork and transport, including air-freight (Jensen et al., 2019), or allow for more targeted recalls, leveraging many opportunities to reduce carbon emissions.

Yet, we still lack reliable information on the detailed energy consumption of different non-PoW blockchains. We also lack information on the quantification on their energy-saving potential for specific use-cases. Together, these remain a field for future work, which will involve a more detailed analysis of the role of consensus and transaction-based overheads and efficiency for a large subset of the consensus mechanisms and blockchain implementations available. It will also involve a discussion about the compromise between the degree of decentralization, security, performance, energy consumption, and further metrics which are of importance for blockchain-based use-cases. Based on such investigations and more reliable numbers, and the development of the most influential blockchain use-cases in practice, one will finally be in a position to decide whether or not the energy consumption of blockchain technology outweighs the savings in a specific scenario.

References

Androulaki, E. et al. (2018). "Hyperledger Fabric: A distributed operating system for permissioned blockchains". In: *Proceedings of the 13th EuroSys Conference*. DOI: 10.1145/3190508.3190538.

Angelis, S. de, L. Aniello, F. Lombardi, A. Margheri, and V. Sassone (2017). *PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain*. URL: https://eprints.soton.ac.uk/415083/2/itasec18_main.pdf.

Beck, R. (2018). "Beyond Bitcoin: The rise of blockchain world". In: *Computer* 51 (2), pp. 54–58. DOI: 10.1109/mc.2018.1451660.

Beck, R., M. Avital, M. Rossi, and J. B. Thatcher (2017). "Blockchain technology in business and information systems research". In: *Business & Information Systems Engineering* 59 (6), pp. 381–384. DOI: 10.1007/s12599-017-0505-1.

Beck, R., C. Müller-Bloch, and J. L. King (2018). "Governance in the blockchain economy: A framework and research agenda". In: *Journal of the Association for Information Systems* 19 (10), pp. 1020–1034. DOI: 10.17705/1jais.00518.

Beincrypto (2020). *Bitcoin's hash rate retraces 40 % this month, slips under 100 EHash/s*. URL: https://beincrypto.com/bitcoins-hash-rate-retraces-40-this-month-slips-under-100-ehash-s/.

Ben-Sasson, E., I. Bentov, Y. Horesh, and M. Riabzev (2019). "Scalable zero knowledge with no trusted setup". In: *Annual International Cryptology Conference*, pp. 701–732. DOI: 10.1007/978-3-030-26954-8_23.

Bitcoin Magazine (2018). *What is the Bitcoin block size limit?* URL: https://bitcoinmagazine.com/guides/what-is-the-bitcoin-block-size-limit.

Buhl, H. U. and M. Jetter (2009). "BISE's responsibility for our planet". In: *Business & Information Systems Engineering* 1 (4), pp. 273–276. DOI: 10.1007/s12599-009-0058-z.

Coinmarketcap (2020). *Top 100 cryptocurrencies by market capitalization*. URL: https://coinmarketcap.com/.

Coinswitch (2019). *Bitcoin mining hardware*. URL: https://coinswitch.co/news/top-10-best-bitcoin-mining-hardware-in-2020-latest-review-and-comparison.

Crosby, M., P. Pattanayak, S. Verma, V. Kalyanaraman, et al. (2016). "Blockchain technology: Beyond Bitcoin". In: *Applied Innovation Review* 2, pp. 6–19. URL: https://j2-capital.com/wp-content/uploads/2017/11/AIR-2016-Blockchain.pdf.

de Vries, A. (2018). "Bitcoin's growing energy problem". In: *Joule* 2 (5), pp. 801–805. DOI: 10.1016/j.joule.2018.04.016.

Dittmar, L. and A. Praktiknjo (2019). "Could Bitcoin emissions push global warming above 2°C?" In: *Nature Climate Change* 9 (9), pp. 656–657. DOI: 10.1038/s41558-019-0534-5.

Douceur, J. R. (2002). "The Sybil attack". In: *International Workshop on Peer-to-Peer Systems*, pp. 251–260. DOI: 10.1007/3-540-45748-8_24.

Eklund, P. W. and R. Beck (2019). "Factors that impact blockchain scalability". In: *Proceedings of the 11th International Conference on Management of Digital EcoSystems*, pp. 126–133. DOI: 10.1145/3297662.3365818.

Eyal, I. and E. G. Sirer (2014). "Majority is not enough: Bitcoin mining is vulnerable". In: *International Conference on Financial Cryptography and Data Security*, pp. 436–454. DOI: 10.1007/978-3-662-45472-5_28.

Fridgen, G., J. Lockl, S. Radszuwill, A. Rieger, A. Schweizer, and N. Urbach (2018a). "A solution in search of a problem: A method for the development of blockchain use cases". In: *Proceedings of the 24th Americas Conference on Information Systems*. AIS, pp. 3460–3469. URL: https://aisel.aisnet.org/amcis2018/StrategicIT/Presentations/14/.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018b). "Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation". In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Georgiadis, E. (2019). *How many transactions per second can Bitcoin really handle? Theoretically*. URL: https://eprint.iacr.org/2019/416.

Gudgeon, L., P. Moreno-Sanchez, S. Roos, P. McCorry, and A. Gervais (2020). “SoK: Layer-two blockchain protocols”. In: *Financial Cryptography and Data Security: 24th International Conference*. Springer, pp. 201–226. DOI: 10.1007/978-3-030-51280-4_12.

Jensen, T., J. Hedman, and S. Henningsson (2019). “How TradeLens delivers business value with blockchain technology”. In: *MIS Quarterly Executive* 18 (4), pp. 221–243. DOI: 10.17705/2msqe.00018.

Kannengießer, N., S. Lins, T. Dehling, and A. Sunyaev (2019). “What does not fit can be made to fit! Trade-offs in distributed ledger technology designs”. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences*, pp. 7069–7078. DOI: 10.24251/hicss.2019.848.

Krause, M. J. and T. Tolaymat (2018). “Quantification of energy and carbon costs for mining cryptocurrencies”. In: *Nature Sustainability* 1 (11), pp. 711–718. DOI: 10.1038/s41893-018-0152-7.

Labazova, O., T. Dehling, and A. Sunyaev (2019). “From hype to reality: A taxonomy of blockchain applications”. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences*, pp. 4555–4564. DOI: 10.24251/hicss.2019.552.

Lockl, J., V. Schlatt, A. Schweizer, N. Urbach, and N. Harth (2020). “Toward trust in Internet of Things (IoT) ecosystems: Design principles for blockchain-based IoT applications”. In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1256–1270. DOI: 10.1109/TEM.2020.2978014.

Mora, C., R. L. Rollins, K. Taladay, M. B. Kantar, M. K. Chock, M. Shimada, and E. C. Franklin (2018). “Bitcoin emissions alone could push global warming above 2°C”. In: *Nature Climate Change* 8 (11), pp. 931–933. DOI: 10.1038/s41558-018-0321-8.

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

O’Dwyer, K. J. and D. Malone (2014). “Bitcoin mining and its energy footprint”. In: *Proceedings of the 25th IET Irish Signals & Systems Conference 2014*, pp. 280–285. DOI: 10.1049/cp.2014.0699.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). “Building a blockchain application that complies with the EU general data protection regulation”. In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Schmidt, N.-H., K. Erek, L. M. Kolbe, and R. Zarnekow (2009). “Sustainable information systems management”. In: *Business & Information Systems Engineering* 1 (5), pp. 400–402. DOI: 10.1007/s12599-009-0067-y.

Stoll, C., L. Klaaßen, and U. Gallersdörfer (2019). “The carbon footprint of Bitcoin”. In: *Joule* 3 (7), pp. 1647–1661. DOI: 10.1016/j.joule.2019.05.012.

Truby, J. (2018). “Decarbonizing Bitcoin: Law and policy choices for reducing the energy consumption of blockchain technologies and digital currencies”. In: *Energy Research & Social Science* 44, pp. 399–410. DOI: 10.1016/j.erss.2018.06.009.

Vranken, H. (2017). “Sustainability of Bitcoin and blockchains”. In: *Current Opinion in Environmental Sustainability* 28. DOI: 10.1016/j.cosust.2017.04.011.

C Research Paper 2 – We need a broader debate on the sustainability of blockchains

Authors:

Alexander Rieger, Tamara Roth, Johannes Sedlmeir, & Gilbert Fridgen

Published in:

Joule (DOI: 10.1016/j.joule.2022.04.013)

Abstract:

-

Keywords:

-

Cryptocurrencies are often criticized not only for their enormous energy consumption and e-waste but also for their carbon emissions, impact on local air quality, and detrimental health effects for humans and animals (de Vries et al., 2022; Gallersdörfer et al., 2020). Criticism ignites especially around the proof-of-work (PoW) consensus mechanism that, for instance, Bitcoin and Ethereum – the two largest cryptocurrencies by market capitalization – use to synchronize and secure their underlying blockchains. This criticism is empirically substantiated and justified but it is often generalized to all blockchains.

As a result, blockchains have gained a negative reputation as environmental polluters even though non-PoW blockchains have comparatively low energy needs and carbon footprints. These blockchains not only warrant a more differentiated analysis but also a discussion about the environmental benefits of blockchain. In fact, there is reason to believe that non-PoW blockchains may enable applications that contribute to sustainability, for instance, by reducing wasteful practices in food supply chains (IBM, 2022), container shipment (TradeLens Collaboration, 2022), and public services (European Commission, 2022) or by facilitating more efficient carbon markets (Gallersdörfer et al., 2022; Gallersdörfer et al., 2020).

In this Commentary, we consequently argue for a broader debate on the sustainability of blockchain. We begin our argument with a discussion of the significant reduction in energy needs possible for public blockchains from using PoW instead of proof-of-stake (PoS). In the second part, we provide measurements for the energy consumption of prominent private blockchains to complement those for major PoW (de Vries et al., 2022; Gallersdörfer et al., 2020) and PoS (Gallersdörfer et al., 2022) blockchains. The third part concludes with a discussion of blockchain applications that may well add to sustainability. Overall, we aim to provide a clearer picture of the energy needs of different blockchains and help to identify areas of application where blockchains can be a source of sustainability.

C.1 Energy efficient public blockchains

The high energy demand of PoW blockchains is rooted in the basic challenge of blockchain networks: ensuring that the blockchain's distributed copies are updated truthfully and reliably. In public settings, the challenge is typically resolved by consensus mechanisms that financially reward network participants for the addition of a truthful new block. The reward can be a certain cryptocurrency balance or/and fees for the transactions included in this block. To guide the election of the network participant who can add the next block, these consensus mechanisms use scarce resources – that is, resources that are costly to replicate. Connecting the probability of being elected to a scarce resource helps public blockchain networks prevent Sybil attacks. With such attacks, adversaries could take control over the network's consensus process. For instance, when all participants in a blockchain network contributed to the consensus mechanism by submitting votes, an attacker could mount a Sybil attack by creating countless dummy participants that outvote honest participants (Sedlmeir et al., 2020).

PoW blockchains are a special – and historically the first – type of public blockchains. As the scarce resource, they use computational power spent on solving cryptographic puzzles, and by extension, "mining" hardware and electric power. Submitting solutions to these puzzles, which are connected to batches of transactions, convinces the other nodes in the blockchain network that a participant has invested the corresponding scarce resource. To keep the number of transactions that a PoW blockchain can process stable, the difficulty of the puzzle automatically adjusts to the amount of computational power in the network. Rising prices of the cryptocurrency, in turn, encourage investments in more computational power, which drives up the puzzle's difficulty and leads to higher energy demand and carbon emissions (de Vries et al., 2022; Gallersdörfer et al., 2020; Sedlmeir et al., 2020). This interdependence means that, for instance in March 2022, Bitcoin has consumed as much electricity as countries like Poland or South Africa (de Vries et al., 2022; Gallersdörfer et al., 2020). It also means that more energy-efficient hardware will not reduce the energy consumption of PoW blockchains in the long run (Sedlmeir et al., 2020).

To avoid this effect, other cryptocurrency networks, like Polkadot and Solana – two of the largest PoS cryptocurrencies by market capitalization – use their cryptocurrency as the scarce resource. These PoS networks require that a certain amount of the cryptocurrency is "put at stake" to be elected to add the next block. In other words, they tie voting power to the amount of cryptocurrency a voter possesses instead of computational power and energy. For some PoS networks, ownership of the cryptocurrency is sufficient for a higher chance at being selected. For others, only locked cryptocurrency balances increase the odds. Locking ensures that the balance cannot be used for a certain time and turns it into a collateral that disincentivizes malicious behavior.

Consequently, the energy needs associated with consensus finding in PoS blockchains are many orders of magnitude smaller than in PoW blockchains. Recent measurements suggest that even the most energy-intensive PoS blockchains require less than 0.002 % of the energy needs of Bitcoin, the most energy-intensive PoW blockchain (Gallersdörfer et al., 2022). In fact, the energy needs of PoS blockchains are comparable to conventional enterprise IT systems; that is, a payment with a PoS cryptocurrency has similar energy requirements as a payment with PayPal (Paypal, 2019) or VISA (VISA, 2019). It is true that these payment systems process significantly more transactions than common PoS blockchains, but their total energy consumption is significantly higher as well. So, when broken down to the transaction level, both types of systems are in fact comparable. Besides significantly lower energy needs, research suggests that PoS can also provide a comparably high level of security as PoW blockchains (David et al., 2018), at least after a phase of fair distribution. Consequently, Ethereum – the cryptocurrency with the currently second largest market capitalization – has decided to switch from PoW to PoS (Ethereum, 2022) and will likely complete this transition in summer 2022.

C.2 Low energy needs of private blockchains

In corporate and government blockchain networks, the number of nodes can be controlled. Moreover, the involved participants know the identities of other participants. That is, they can associate the public keys of the blockchain nodes with an organization or individual. In such ‘private’ networks, identity can act as the scarce resource and enable consensus mechanisms that build on ‘one participant, one vote’ or reputation-weighted voting. Like PoS, these ‘identity-based’ consensus mechanisms do not require the competitive solving of cryptographic puzzles to resist Sybil attacks.

In Figure 1, we present measurements of these needs for a selection of popular private blockchains. Specifically, we selected blockchains that are both used extensively in industry and government projects and that have been subjected to performance analyses in the academic literature. For our measurements, we deployed theses blockchains on Amazon Web Services, where each node ran on a separate virtual machine. We then measured the virtual machines’ resource utilization for different throughput levels between 1 tx/s and the respective networks’ maximum capacity. From these resource utilizations, we derived power consumption levels. Specifically, we first checked that there is a strong linear relation between transaction throughput and marginal power consumption levels. That is, we verified that energy consumption increased with the number of processed transactions. We then calculated the values presented in Figure 1 as the average over the different throughput levels. The error bars in the main panel represent the standard deviation over these averaged levels.

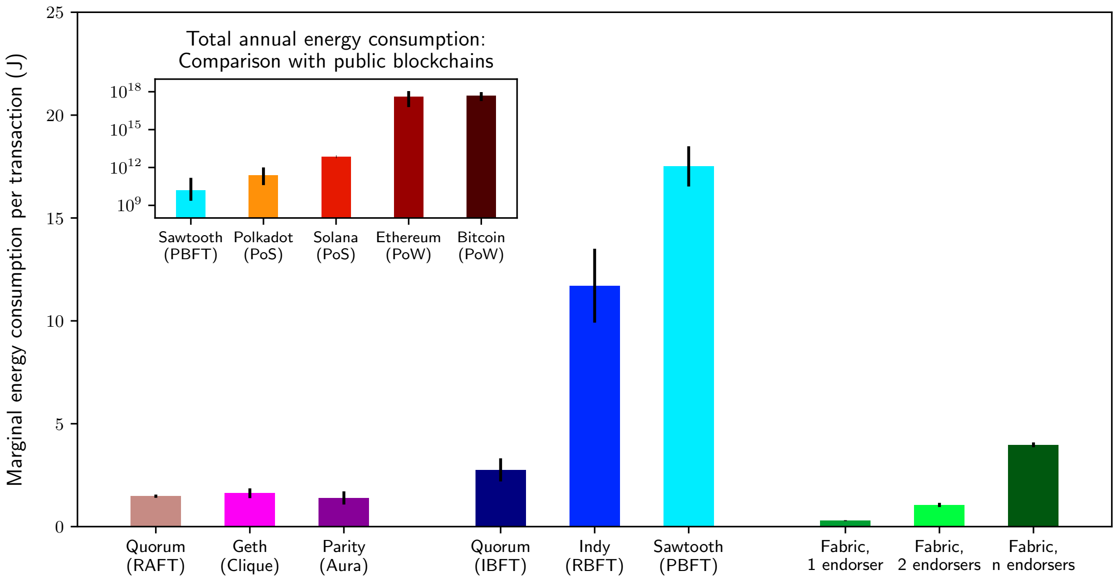


Figure 1: Marginal energy consumption per transaction for selected private blockchains (network size of 32 nodes). The "marginal energy per transaction" values in the main panel exclude idle consumption. We chose a network size of 32 nodes for the panel as this size is representative of many larger private networks, such as the European Blockchain Services Infrastructure (European Commission, 2022). See the Supplemental information for details on the underlying calculations of the main panel. The small panel in the top-left corner offers a comparison against selected public blockchains on a "total annual energy consumption" basis. It applies a logarithmic scale. For the public PoS blockchains, we used measurements by the Crypto Carbon Ratings Institute for Polkadot and Solana (Gallersdörfer et al., 2022). Polkadot and Solana are the public PoS blockchains with the smallest and largest "total annual energy consumption" among the six public PoS blockchains with the highest market capitalization (Gallersdörfer et al., 2022). For the public PoW blockchains, we used Digiconomist values to calculate lower bounds and best guesses for Ethereum and Bitcoin (Digiconomist, 2022), as well as current cryptocurrency prices, transaction fees, and a lower bound of 0.05 USD per kWh of electricity for their upper bounds (de Vries et al., 2022; Gallersdörfer et al., 2020). We illustrate these lower and upper bounds with the error bars in the small panel. Ethereum and Bitcoin are the public PoW blockchains with the highest market capitalization and energy consumption.

Figure 1 highlights that private blockchains, like public PoS blockchains, have low energy needs. These needs naturally increase with network size and tend to grow with the required level of resilience to failure and attack (Figure 2). Yet, total energy consumption remains low even for high transaction throughput because most private blockchain networks are comparatively small due to performance, data privacy and data separation considerations. In essence, private blockchain networks are just a small collection of servers that host a shared database.

The interpretation of Figures 1 and 2 requires some caveats. The marginal energy consumption per transaction metric is useful for non-PoW blockchains in which transaction processing can represent a major share of the overall energy needs. However, it is not perfect as 'idle' consumption can also present a sizeable share for these blockchains (Sedlmeir et al., 2020). Moreover, it should not be used for PoW blockchains

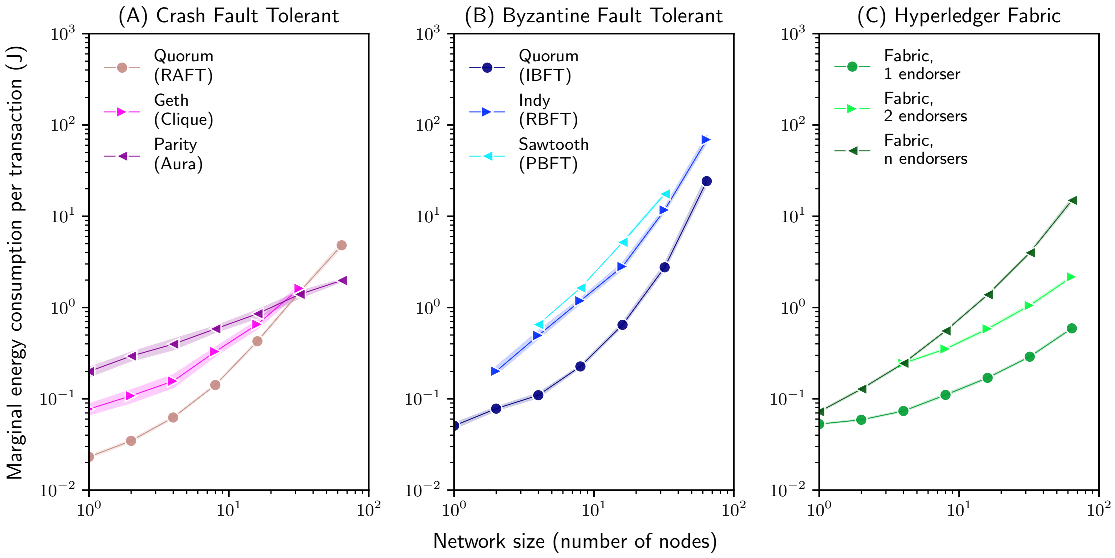


Figure 2: Scaling behavior of the marginal energy consumption per transaction for selected private blockchains. Private blockchains have low energy needs - irrespective of their tolerance to faults and manipulations. The consensus mechanisms in Panel I (Crash Fault Tolerant) are resistant to a certain number of faulty nodes. The mechanisms in Panel II (Byzantine Fault Tolerant) can additionally cope with a certain number of malicious nodes. Hyperledger Fabric networks (Panel III) are resistant to failure and certain attacks. See the supplemental information for more details on the consensus mechanisms and underlying calculations.

in which overall energy consumption is largely independent of the number and complexity of processed transactions. That is, a higher number and complexity of transactions, such as for the creation of NFTs, would not increase the total power consumption of PoW blockchains in a meaningful way (Sedlmeir et al., 2020). Slightly elevated energy needs are nevertheless possible due to increased cumulative transaction fees and a higher cryptocurrency price as a result of popularity gains.

C.3 Sustainability with blockchain

While the debate on energy consumption, e-waste, and other environmental and health impacts of blockchain is extensive (de Vries et al., 2022; Gallersdörfer et al., 2022; Gallersdörfer et al., 2020; Sedlmeir et al., 2020), potential benefits are often marginalized. This is surprising because companies and governments increasingly use blockchain applications that could contribute to sustainability. For instance, blockchain has gained traction for sustainable supply chain management, where its use can ensure increased efficiency and prevent unnecessary waste and surplus production. IBM FoodTrust is a prominent example (IBM, 2022). IBM created FoodTrust in collaboration with major retailers such as Walmart and Unilever to enable extensive product monitoring across supply chains and to prevent fresh produce from being disposed of due to insufficient monitoring. This, in turn, can boost the sustainability of food supply chains. Other blockchain applications enable the digitalization of previously paper-based processes, such as TradeLens (TradeLens Collaboration, 2022). TradeLens was developed by IBM and Maersk, the world's largest container shipping company, to reduce paper- and often airmail-based data exchange in container shipment.

Even if we assume that these private blockchain applications were completely powered by coal (average 2020 US emission factor for coal: 1.01 kg or 2.23 pounds CO_2 eq per kWh (US Energy Information Administration, 2022)), this translates into a carbon footprint of 2.81×10^{-7} kg CO_2 eq for each Joule. In comparison to the possible carbon savings, this value is marginal. For instance, it would be enough if one FoodTrust transaction helped to avoid the disposal of one gram of field vegetables (estimated carbon footprint of 3.30×10^{-4} kg CO_2 eq (Petersson et al., 2021)) or if one TradeLens transaction shortens the voyage time of a container ship by a thousandth of a second (estimated 2018 carbon footprint of international shipping: 1.33 kg CO_2 eq per s (International Maritime Organization, 2020)). Of course, these estimates are subject to some degree of uncertainty and the total CO_2 footprint of private blockchains may be higher – for instance due to the additional footprint of the underlying hardware. Yet, it is unlikely that more precise estimates will add the several orders of magnitude required to offset possible savings. In effect, there is growing indication that companies and governments can contribute to the sustainability of supply chains with blockchain, not despite blockchain.

Naturally, using blockchain for increased sustainability is not limited to supply chain management. Similar efforts to reduce inefficiencies in public administration are under way with the European Union's European Blockchain Services Infrastructure (European Commission, 2022). Blockchain technology is also frequently discussed as a key to more efficient carbon markets (Al Sadawi et al., 2021). Overall, the use of blockchain technology could contribute to sustainability in areas where it can (1) make processes more efficient, (2) replace the paper-based exchange of sensitive information, or (3) reduce the use of fossil fuels or loss of produce, and where the environmental costs of using blockchain do not exceed sustainability benefits (European Commission, 2022; IBM, 2022; TradeLens Collaboration, 2022).

C.4 Conclusion

Given the broad range of blockchains beyond PoW, we argue for a more differentiated debate about the sustainability of blockchain technology. We particularly caution against blindly extending the critique of PoW to PoS and private blockchains, which both have low energy needs. Since some of them may even add to sustainability, we also see a need for a more balanced debate that goes beyond environmental costs and reflects on environmental benefits. This debate can build on ongoing efforts to identify areas of application in which blockchain could contribute to sustainability (Sedlmeir et al., 2020). Moreover, it can add to a comprehensive overview of reference projects, their benefits and costs, and the consensus mechanisms used.

Standardization bodies could also make an important contribution to differentiation and balance with a carbon accounting framework for blockchain applications. With such a framework, companies could evaluate different blockchain designs and hosting options and establish the corresponding net carbon emissions. Moreover, such a framework would allow auditors to certify the sustainability of blockchain applications. A promising starting point can be established frameworks for corporate carbon accounting.

Declaration of Interests

The authors declare the following competing interests: A.R., T.R., and G.F. are supported by PayPal and the Luxembourg National Research Fund FNR (P17/IS/13342933/PayPal-FNR/Chair in DFS/ Gilbert Fridgen). PayPal's financial support is administered via the FNR, and by contractual agreement, PayPal has no involvement in A.R., T.R., and G.F.'s research.

References

Al Sadawi, A., B. Madani, S. Saboor, M. Ndiaye, and G. Abu-Lebdeh (2021). "A comprehensive hierarchical blockchain system for carbon emission trading utilizing blockchain of things and smart contract". In: *Technological Forecasting and Social Change* 173, p. 121124. DOI: 10.1016/j.techfore.2021.121124.

David, B., G. P, A. Kiayias, and A. Russell (2018). "Ouroboros Praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain". In: *Advances in Cryptology – EUROCRYPT 2018*. Ed. by J. Nielsen and V. Rijmen. Springer. DOI: 10.1007/978-3-319-78375-8_3.

de Vries, A., U. Gallersdörfer, L. Klaaßen, and C. Stoll (2022). “Revisiting Bitcoin’s carbon footprint”. In: *Joule* 6 (3), pp. 498–502. DOI: 10.1016/j.joule.2022.02.005.

Digiconomist (2022). *Bitcoin energy consumption index and Ethereum energy consumption index*. URL: https://digiconomist.net/.

Ethereum (2022). *Ethereum. The Beacon Chain*. URL: https://ethereum.org/en/eth2/beacon-chain/.

European Commission (2022). *European Blockchain Services Infrastructure*. URL: https://ec.europa.eu/digital-building-blocks/wikis/display/EBSI/Home.

Gallersdörfer, U., L. Klaaßen, and C. Stoll (2022). *Energy efficiency and carbon footprint of proof of stake blockchain protocols*. URL: https://www.carbon-ratings.com/dl/pos-report-2022.

Gallersdörfer, U., L. Klaaßen, and C. Stoll (2020). “Energy consumption of cryptocurrencies beyond Bitcoin”. In: *Joule* 4 (9), pp. 1843–1846. DOI: 10.1016/j.joule.2020.07.013.

IBM (2022). *IBM Food Trust*. URL: https://www.ibm.com//blockchain/solutions/food-trust.

International Maritime Organization (2020). *Fourth greenhouse gas study*. URL: https://www.imo.org/en/OurWork/Environment/Pages/Fourth-IMO-Greenhouse-Gas-Study-2020.aspx.

Paypal (2019). *2019 global impact report*. URL: https://www.paypalobjects.com/marketing/web/us/globalimpact/PayPal_2019_Global_Impact_Report_FINAL.pdf.

Petersson, T., L. Secondi, A. Magnani, M. Antonelli, K. Dembska, R. Valentini, A. Varotto, and S. Castaldi (2021). “A multilevel carbon and water footprint dataset of food commodities”. In: *Scientific Data* 8 (1). DOI: 10.1038/s41597-021-00909-8.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020). “The energy consumption of blockchain technology: Beyond myth”. In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

TradeLens Collaboration (2022). *Tradelens*. URL: https://www.tradelens.com/.

US Energy Information Administration (2022). *How much carbon dioxide is produced per kilowatthour of U.S. electricity generation*. URL: https://www.eia.gov/tools/faqs/faq.php?id=74&t=11.

VISA (2019). *2019 corporate responsibility & sustainability report*. URL: https://usa.visa.com/dam/VCOM/download/corporate-responsibility/visa-2019-corporate-responsibility-report.pdf.

C.5 Supplemental information

Figure 1

The values in Figure 1 are based on measurements with the Distributed Ledger Performance Scan (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021; Sedlmeir et al., 2021), an open-source framework for determining performance characteristics of various blockchain technologies. With the DLPS, we deployed blockchain networks as a cluster of virtual machines (one node per virtual machine) on Amazon Web Services. Each of the nodes in our setup had 4 virtual cores (corresponding to 2 physical cores) and 8 GB of RAM. This configuration yields a good tradeoff between throughput and costs for many of the examined blockchains3. Moreover, we set up 'client' virtual machines to broadcast transactions to the cluster of blockchain nodes. We then measured the nodes' resource utilization (CPU and memory) for different rates of 'throughput' (transactions per second sent from the client virtual machines).

To translate resource utilization into power consumption levels, we used available power consumption estimates for Amazon Web Services' EC2 server architecture, which is based on Intel Xeon 8175 processors with 48 physical cores (Davy, 2018). Specifically, we assumed around 100 W for idle consumption up to 550 W for maximum CPU and memory usage. These estimates are in line with those used by Gallersdörfer et al. (2022) to calculate the energy needs of PoS networks. For instance, Gallersdörfer et al. (2022) estimate that a Solana node consumes around 80 W for idle consumption and 220 W under average load. Manufacturer specifications for the hardware they used, in turn, suggest 280 W for maximum CPU usage.

As data centers typically use a single processor for multiple virtual machines, we calculated the virtual machines' idle consumption by proportionally attributing the idle consumption of the physical processor. More specifically and considering an idle consumption of around 100 W for the Intel Xeon 8175 socket with 48 physical cores, we calculated an idle consumption of $\frac{4}{96}$ times 100 W for each of our virtual machines with 4 virtual cores. In a second step, we did the same attribution for 100 % CPU and memory utilization respectively. In the third and last step, we interpolated the power needs for 0 % and 100 % resource utilization, which gave as a function to translate resource utilization into power consumption levels.

To increase robustness, we conducted our measurements for different throughput rates and conducted each measurement three times (sending requests with a fixed throughput rate for 20 seconds to a blockchain network with 32 nodes). We chose 20 seconds to balance costs and reliability of our measurements. We also conducted several spot-checks with measurements over 5 minutes to make sure that there are no long-term effects that negatively affect throughput, such as congestion or accumulating memory consumption. After validating that resource utilization was approximately linearly dependent on throughput, we then worked with averaged values for Figure 1.

Figure 2

Networks with a crash-fault tolerant consensus mechanism have energy needs that scale approximately quadratically. Those networks with more secure, byzantine-fault tolerant consensus mechanisms scale approximately cubically. The scaling behavior of Hyperledger Fabric networks depends on their endorsement policies; that is, their energy needs scale approximately quadratically if an increasing number of nodes is required for endorsement, otherwise they increase approximately linearly.

All values in Figure 2 are again based on DLPS (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021; Sedlmeir et al., 2021) measurements of CPU and memory utilization for virtual machines with 4 virtual cores and 8 GB RAM on Amazon Web Services as well as the same estimation approach that we used for Figure 1. For those consensus mechanisms that require a certain minimum network size or become unstable beyond a certain size, we report only a subset of measurements. It is worth noting that our calculations for those mechanisms that can work with only one node indicate very little energy needs for consensus in small networks.

References

Davy, B. (2018). *Estimating AWS EC2 instances power consumption.* URL: https://medium.com/teads-engineering/estimating-aws-ec2-instances-power-consumption-c9745e347959.

Fraunhofer FIT and Universities of Bayreuth and Luxembourg (2021). *Distributed ledger performance scan repository.* URL: https://github.com/DLPS-Framework.

Gallersdörfer, U., L. Klaaßen, and C. Stoll (2022). *Energy efficiency and carbon footprint of proof of stake blockchain protocols.* URL: https://www.carbon-ratings.com/dl/pos-report-2022.

Sedlmeir, J., P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021). “The DLPS: A new framework for benchmarking blockchains”. In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

D Research Paper 3 – The DLPS: A new framework for benchmarking blockchains

Authors:

Johannes Sedlmeir, Philipp Ross, Andre Luckow, Jannik Lockl, Daniel Miehle, & Gilbert Fridgen

Published in:

Proceedings of the 54th Hawaii International Conference on System Sciences
(DOI: 10.24251/hicss.2021.822)

Abstract:

Distributed Ledger Technologies (DLT) promise to revolutionize business ecosystems by permitting secure transactions without intermediaries. A widely recognized challenge that inhibits the uptake of DLT is scalability and performance. Hence, quantifying key metrics such as throughput and latency is crucial for designing DLT-based infrastructures, applications, and ecosystems. However, current benchmarking frameworks for blockchains[1] do not cover the whole benchmarking process; impeding transparent comparisons of different DLT networks. In this paper, we present the distributed ledger performance scan (DLPS), an open-source[2] framework for end-to-end performance characterizations of blockchains, addressing the need to transparently and automatically evaluate the performance of highly customizable configurations. We describe our new framework and argue that it significantly improves existing DLT benchmarking solutions. To demonstrate the capabilities of the DLPS, we also summarize the main results obtained from a series of experiments that we have conducted with it, giving a first comprehensive comparison of essential scalability properties of several commonly used enterprise blockchains.

Keywords:

Blockchain engineering, benchmarking, blockchain, distributed ledger technologies, performance, scalability

[1] Strictly speaking, blockchains are a subset of distributed ledger technology (DLT), but in this work, we will use the terms interchangeably since all distributed ledgers we investigated are blockchains.

[2] The DLPS is available at https://github.com/DLPS-Framework.

D.1 Introduction

DLT are expected to play an important role in tomorrow's IT landscape (Iansiti and Lakhani, 2017). Nakamoto (2008) introduced the first blockchain, Bitcoin, in 2008 and established a peer-to-peer (P2P) digital currency without the need for trusted intermediaries such as banks. Buterin et al. (2014) then extended the scope of blockchain technology from financial transactions to the execution of general process logic and implemented respective capabilities in Ethereum. This finally realized a vision first communicated by Szabo (1997), where so-called smart contracts could be concluded digitally and on a P2P basis, without any trusted intermediary. Since then, a large number of blockchain-based use cases have emerged, outreaching the financial sector (Jensen et al., 2019; Rieger et al., 2019).

To secure a distributed ledger (DL) without a distinguished administrator against malicious behavior, storing data and performing operations on the ledger is performed *redundantly* on all participating nodes. A suitable tamper-sensitive data structure (often Merkle-trees) and usage of public-key cryptography make retrospective manipulations easily detectable and allow for secure authentication (Butijn et al., 2020). A *consensus mechanism*, a mixture of economic incentives and cryptographic methods, ensures that the presupposed benevolent majority agrees on which transactions to operate. Redundancy and the additional overhead caused by the consensus mechanism, however, lead to a significantly decreased performance of DLs when compared to centrally managed databases (Butijn et al., 2020). This makes building decentralized applications very challenging as established DLT networks usually cannot elastically scale on demand (Poon and Dryja, 2016). Therefore, an in-depth understanding of DLT performance becomes essential, as the performance poses a key aspect for the viability of emerging decentralized applications.

To address the performance requirements of enterprise blockchain solutions, permissioned DLs have been developed. They restrict participation, allowing for other types of consensus mechanisms that generally exhibit finality and lower latency. Moreover, in an enterprise scenario, hardware and bandwidth restrictions are less relevant than in a permissionless system. However, enterprise IT-systems must also meet high performance requirements, and throughput of permissioned DLT still lags significantly behind that of their centralized counterparts. Consequently, research considers performance a major obstacle for productive usage of enterprise DLT implementations (Vukolić, 2016).

Unfortunately, literature only provides limited knowledge regarding the performance of enterprise blockchain solutions, and for the few currently available results, we also discovered quite different performance results. Moreover, blockchain implementations have already become heterogeneous and are quickly evolving, so no generally acknowledged benchmarking tool has been established to comprise all of these particularities. Moreover, existing work on benchmarking does not provide clear definitions of key metrics such as throughput and latency, and do not specify the algorithms that they use to measure these key metrics, which leads to a lack of transparency and reproducibility.

In this paper, we address this research gap by presenting a transparent and highly flexible, open-source framework for obtaining reliable performance data of several enterprise DLT solutions. It was implemented in an iterative approach within an enterprise project that needed reliable performance comparisons to support the choice of enterprise blockchain technology for their use case. We argue that the DLPS covers the deficiencies of existing approaches and allows to measure well-defined quantitative key performance indicators of different DLT with a universal, comprehensive and transparent benchmarking algorithm. We also present and discuss the results of a first systematic scalability comparison of the DLT that we have already integrated to compare our framework with previous solutions and to demonstrate that the DLPS is applicable to a variety of technologies. To the best of our knowledge, the range of investigated DLT and also the variety of network sizes that we tested is, so far, unique.

The remainder of this paper is structured as follows: Sec. D.2 gives an introduction to essential background concepts and presents some of the permissioned DLT that we have already integrated into the DLPS. Sec. D.3 provides an overview of existing work on benchmarking permissioned DLT and sketches their main shortcomings. We then introduce the DLPS in Sec. D.4. In Sec. D.5, we present and discuss the main results of our exemplary scalability and performance analyses in order to demonstrate the capabilities of the DLPS. We conclude with a summary and our plans for future work in Sec. D.6.

D.2 Background

D.2.1 Consensus mechanisms

For permissionless blockchains, which constitute the technology behind cryptocurrencies, the most common consensus mechanisms are proof-of-work (PoW) and proof-of-stake (PoS). They tie voting power in the system to some scarce resource – energy in PoW and capital in PoS – to defend the system against Sybil attacks. These consensus mechanisms generally exhibit high latency and do not provide finality, implying that even after some nodes have performed a particular transaction, one has to wait minutes to hours before the probability that this transaction will be replaced is sufficiently small (Gervais et al., 2016). Moreover, PoW is very energy-intensive (Sedlmeir et al., 2020). For permissioned blockchains, voting-based consensus mechanisms are applicable because participation in consensus is restricted. These consensus mechanisms provide finality and also much lower latency, but are only viable for small-scale networks.

In most voting-based consensus mechanisms, the participants (i.e., nodes) usually agree on a common leader, which proposes new transactions and distributes them to the other nodes called followers. In a consensus mechanism that exhibits **crash fault tolerance (CFT)**, the other nodes will realize a crash of their leader and elect a new leader. However, the followers blindly rely on their leader as long as it is active, so a *malicious* leader can be problematic. Prominent examples for a CFT consensus mechanism are Kafka and RAFT (Ongaro and Ousterhout, 2014). Since leader election needs a majority vote, a network must be of at least size 2f+1 to handle f crashing nodes.

By contrast, consensus mechanisms with **Byzantine fault tolerance (BFT)** can deal not only with crashes but also with arbitrary malicious behavior. Like CFT protocols, an elected leader proposes new blocks, while multiple cross-checks ensure consistency among the non-faulty nodes. Therefore, the communication overhead of a BFT protocol grows faster in the number of nodes than for a CFT protocol. In general, the best case accomplishable is that a network of size 3f+1 can deal with f malicious nodes (Lamport et al., 1982). Popular implementations are practical BFT (PBFT) (Castro, Liskov, et al., 1999), Istanbul BFT (IBFT) (Saltini, 2019), and redundant BFT (RBFT) (Aublin et al., 2013). RBFT is an advancement of PBFT which offers very reliable performance also under the actual presence of malicious behaviour (Aublin et al., 2013).

Proof-of-authority (PoA) consensus mechanisms have been implemented to achieve an approximation to CFT or BFT at less overhead. Prominent examples thereof are Clique, and Aura (Angelis et al., 2017). They generally use a simplified leader election and leave out different steps thereafter as compared to PBFT protocols. In Angelis et al. (2017), the authors question whether these consensus mechanisms are adequate for blockchains because they cannot guarantee data consistency among all non-faulty nodes (known as *safety*).

DLT	Consensus	SC Languages	Version
Eth. (Geth)	PoA (Clique)	Solidity	1.9.8
Eth. (Parity)	PoA (Aura)	Solidity	2.5.10
Fabric	Solo, Kafka, RAFT	Go, Javascript	1.4.4
Indy	RBFT	-	1.12.0
Quorum	RAFT, IBFT	Solidity	2.3.0
Sawtooth	RAFT, PBFT, PoET,	Go, Python, ...	1.2

Table 1: Comparison of the DLT that we integrated in the DLPS and evaluated in the experiments.

A large variety of other consensus mechanisms exists, but so far, these have had only little adoption. One that should be mentioned in this paper is proof of elapsed time (PoET), which uses trusted hardware (the Intel SGX) to establish tamper-proof random number generation for nodes, which then determines who may publish the next block. It claims to offer solid performance even in permissioned networks with a larger number of validators (Shi et al., 2019).

D.2.2 Permissioned blockchains

We now give a short overview of the permissioned blockchain systems currently integrated in the DLPS. All of them are open-source, and – apart from Indy – provide means to implement Turing-complete transaction logic, also know as smart contracts. Table 1 summarizes these DLT and some of their characteristics at the time that we conducted our experiments presented in Sec. D.5.

Ethereum was the first public blockchain which supported smart contracts, enabling guaranteed and tamper-proof execution of program code (Christidis and Devetsikiotis, 2016) in the so-called Ethereum Virtual Machine (EVM). It is developed by the Ethereum foundation. While the popular public chain currently uses PoW, **Private Ethereum Networks** have been developed on which one can capitalize on other consensus mechanisms. The two most popular Ethereum clients for private networks on which we focus in this work, Geth and Parity, use the PoA consensus mechanisms Clique and Aura respectively (Angelis et al., 2017).

Fabric is a framework for building private permissioned blockchains. Fabric is special among other DLT architectures for one main reason: Most blockchains (both permissionless and permissioned ones) use a so-called validate-order-execute paradigm (Androulaki et al., 2018): They first check whether a transaction is legitimate (*validate*), then agree on the transactions and their sequential arrangement in the next block through consensus (*order*), and finally apply the transactions on their local ledger through the blockchain's state transition function (*execute*). By contrast, Fabric entails an execute-order-validate paradigm: At first, according to a so-called *endorsement policy*, a subset of the nodes simulates the outcome of performing a transaction and signs it (*execute*). The client collects the required endorsements (specified by the smart contract (SC) that, for example, three out of five nodes need to agree) and hands them to the *ordering service*, which collects the transactions, checks whether the endorsement policy is satisfied, puts them in blocks, and distributes the blocks to all nodes (*order*). Finally, when nodes apply the transactions to their ledger, they have to check for read-write collisions as simulations are not necessarily ordered (*validate*). By this design, the degree of redundancy can be customized according to the needs of each SC (Androulaki et al., 2018). Fabric currently offers 3 different kinds of consensus: Solo (i.e., a single ordering node, mainly intended for testing purposes), Kafka, and RAFT. It also supports different databases for the transaction log and the current world state, namely, LevelDB and CouchDB (*Hyperledger Fabric Repository* 2022).

Indy is a public permissioned blockchain. Participation in consensus is thus restricted while read access is not. Indy is developed mainly for specific purposes in identity management and hence optimized for reading operations because they will occur much more frequently. Therefore, all transactions are signed by all nodes and include a timestamp such that querying from a single node is still sufficient to rule out undetected malicious answers. Indy runs Plenum as a consensus mechanism, which is a slightly adapted version of RBFT. In contrast to all the other blockchains presented here, Indy does not support arbitrary smart contracts, but only a basic set of transactions related to identity management (*Hyperledger Indy repository* 2020).

Quorum is a private permissioned blockchain project led by J.P. Morgan. It originates from Ethereum but aims to allow for business applications that are not feasible on the public main net due to performance restrictions. Quorum supports RAFT and IBFT consensus mechanism (Baliga et al., 2018).

Sawtooth is another permissioned blockchain project similar to Fabric, Sawtooth separates between the application and core system level, allowing using different programming languages for SC development. Sawtooth supports multiple consensus mechanisms, namely RAFT, PBFT, and PoET, which one can even switch at runtime (Shi et al., 2019).

D.3 Related work

The performance of permissionless blockchains can be monitored by analyzing publicly accessible data, and their architecture is not customizable for a specific use case. Consequently, there is only limited need to conduct benchmarking in the context of use case technology selection and optimization. By contrast, benchmarks for permissioned DLT are desperately needed for enterprises in designing decentralized applications. Originally, we intended to collect available performance results or existing benchmarking frameworks to decide for a specific permissioned network in an enterprise project. For this purpose, we conducted a literature research for the search string "(blockchain OR distributed ledger technology) AND (performance OR throughput OR latency) AND (benchmarking OR measurement OR evaluation OR analysis)" on the Google Scholar, ACM DL, and IEEE Explore databases. We found that there are two existing frameworks, and most articles that study the performance of specific blockchains refer to one of these.

The first systematic benchmarking framework for permissioned blockchains was Blockbench. It relies on established YCSB and Smallbank benchmarks and integrates private Ethereum (Geth and Parity), Fabric, and Quorum. The framework is open-source and modular and provides smart contracts to evaluate different workloads. Nevertheless, beyond reacting to some blockchain-related updates, there have not been significant advancements since 2017. Dinh et al. (2017) use Blockbench for an in-depth comparison of the performance of Geth, Parity, and a by 2019 outdated version of Fabric.

The other prominent available framework is Caliper. It was originally developed to benchmark only DLT of the Hyperledger project, but now also integrates Ethereum-based DLT. Caliper contains different basic SCs, which trigger particularly CPU- or i/o-heavy transactions. These experiments are valuable for grasping different characteristics of performance.

In our literature review, we found that while there have been valuable performance measurements on various permissioned DLT, the data is highly fragmented over various contributions, and none of the papers we encountered gives a full description of their benchmarking process or setup. Consequently, the lack of a fully transparent description of how performance metrics were obtained leads to a serious lack of comparability across different works. Fan et al. (2020) structure some of the related work that we found in our literature review, and already from this subset it gets evident that benchmarking data is highly fragmented across multiple works, the results generally vary significantly, and it is particularly to compare the results. In our opinion, the reason is that yet no benchmarking framework is sufficiently standardized to provide comparability, highly customizable, and at the same time, ease of use. For example, none of the publications from our literature review provides a precise description of the overall definitions and assumptions underlying the measurements of the key performance indicators throughput and latency. Also, they leave the setup of blockchain and client nodes to the user, which both raises the hurdle to start benchmarking and also impairs comparability because different benchmarks happen on different infrastructures. Integrating several DLT in a single framework is a huge challenge because blockchain technologies are quickly evolving, which often implies shortcomings in the documentation, stability issues, and difficulties in getting familiar with the technology and starting a functioning test network. This is particularly important when conceptualizing enterprise blockchain architectures, in which parameters such as the number of nodes or the block-time could be tuned according to requirements, and a tool that allows testing the performance for different choices of parameters would make things much easier for the engineers.

Driven by the motivation to improve DLT benchmarking and build a standardized framework that is easily accessible and useful to a broad community, as well as to obtain reliable comparisons of DLT performance for enterprises striving to adopt DLT solutions, we took this challenge and implemented the DLPS as an end-to-end pipeline with fully automatic node and client setup, benchmarking, and evaluation. This brings the advantage of built-in scalability of the network size in experiments, as well as an easy setup for researchers and practitioners who are not experts in each of the integrated DLT. At every point, we implemented the blockchain network in the way suggested in the respective development repository, with a client-node architecture that seemed close to how one would implement it in reality (e.g., the provided SDK or popular software such as web3 for Ethereum-like blockchains). Inspired by the functionality of Caliper, we have also integrated different workloads. Currently, we offer doNothing (empty workload), writeData (writing a single key-value pair), matrixMultiplication (CPU-heavy workload), and writeMuchData (i/o- heavy workload).

While the benchmarking process itself is standardized, the blockchain, node, and SC functionalities are highly configurable through a single config file, thus providing highly customizable workloads and configurations while maintaining a standardized benchmarking process. By defining and implementing the intuitive benchmarking logic as described in Sec. D.4 and using a realistic client-node setup as well as developing representative workloads that capture the characteristics of many real-world use cases, we also naturally adopt standard best practices for computer systems evaluation (Jain, 1991; Ousterhout, 2018).

D.4 The distributed ledger performance scan

D.4.1 Definition of key metrics

Following most of the related work referenced in Sec. D.3, we focus on the key performance indicators *throughput* and *latency*. Since we could not find a clear definition of these metrics in any of the related work, we start with developing precise definitions, on top of which we can implement a generic algorithm to measure them.

Generally speaking, if we send requests at a certain frequency f_{req} to a DL, this will result in a corresponding response rate $f_{\mathrm{resp}} \equiv f_{\mathrm{resp}}(f_{\mathrm{req}})$ of successfully performed transactions. We define *maximum sustainable throughput* $\hat{f}$ as the maximum reachable rate f_{resp} which the blockchain can reliably process over a longer period (e.g., one minute) when we try different request rates f_{req}:

$$\hat{f} := \max\{f_{\mathrm{resp}}(f_{\mathrm{req}}) : f_{\mathrm{req}} \geq 0\}. \tag{3}$$

Latency l is generally defined as the average time between sending a request and receiving confirmation that it was operated on a sufficient number of nodes (e.g., on at least 2/3 of all nodes). This quantity may depend on the load which the system is currently facing, so $l \equiv l(f_{\mathrm{req}})$. We define *latency (at stress)* as latency at precisely the request rate at which we attain maximum sustainable throughput (see (3)):

$$\hat{l} := l(\hat{f}_{\mathrm{req}}) \qquad \text{where} \qquad f_{\mathrm{resp}}(\hat{f}_{\mathrm{req}}) = \hat{f}. \tag{4}$$

The approach which we have just sketched relies on a few assumptions. For example, we assume that throughput depends only on f_{req}, while a real system will exhibit time-dependent fluctuations. Furthermore, $\hat{f}_{\mathrm{req}}$ is not well-defined if f_{resp} does not have a unique maximum. However, the results of our experiments, which we describe later, suggest that actually, the reality is quite close to our simplification, and – not surprisingly – that $\hat{f} \approx \hat{f}_{\mathrm{req}}$. The highest f_{resp} is thus achieved when the request rate matches the maximum sustainable throughput.

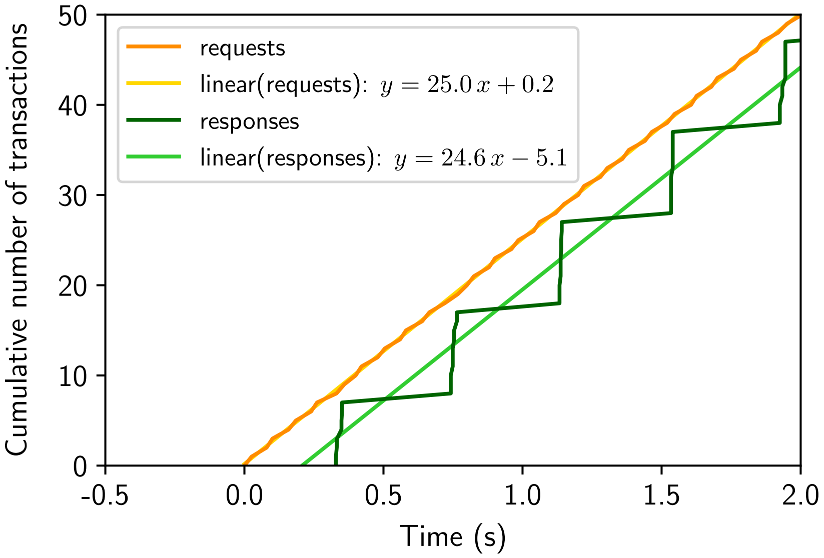


Figure 1: Sending requests to a DLT and getting responses.

To measure $\hat{f}$, we implemented the following algorithm: We start sending requests from some clients to some blockchain nodes at a fixed total frequency f_{req} for a certain duration. For each of the asynchronous requests, we record both the time the client sends it and the time the associated response confirming its execution arrives at the client. Fig. 1 displays data obtained from sending requests to a small Fabric network. It illustrates that by plotting the cumulative number of requests resp. responses against time, we can define f_{req} and f_{resp} as the slope of their corresponding linear regressions: Indeed, differences Δy on the y-axis in a period Δx correspond to the number of transactions sent resp. completed in this period, so the slope $\frac{\Delta y}{\Delta x}$ is precisely the corresponding request resp. response rate. This definition is very robust because it is insensitive to a few outliers. In the picture, we observe that $f_{\text{req}} \approx f_{\text{resp}}$. Note also that in Fig. 1, responses are received in batches of around 10 transactions, representing new, confirmed blocks.

As long as the linear regressions of request and response curves are parallel (i.e., $f_{\text{resp}}(f_{\text{req}}) \approx f_{\text{req}}$), the average time between sending and receiving a transaction is given by the shift between the intercepts of the two regressions with the x-axis (this is a short computation). In Fig. 1, latency is therefore given by approx. 0.2 s. However, if $f_{\text{resp}}(f_{\text{req}}) \not\approx f_{\text{req}}$, due to a potentially growing queue, the former definition of latency in terms of average delay depends on the duration of the experiment while the latter does not. We therefore define latency $l(f_{\text{req}})$ as the shift between the x-intercepts of the regressions

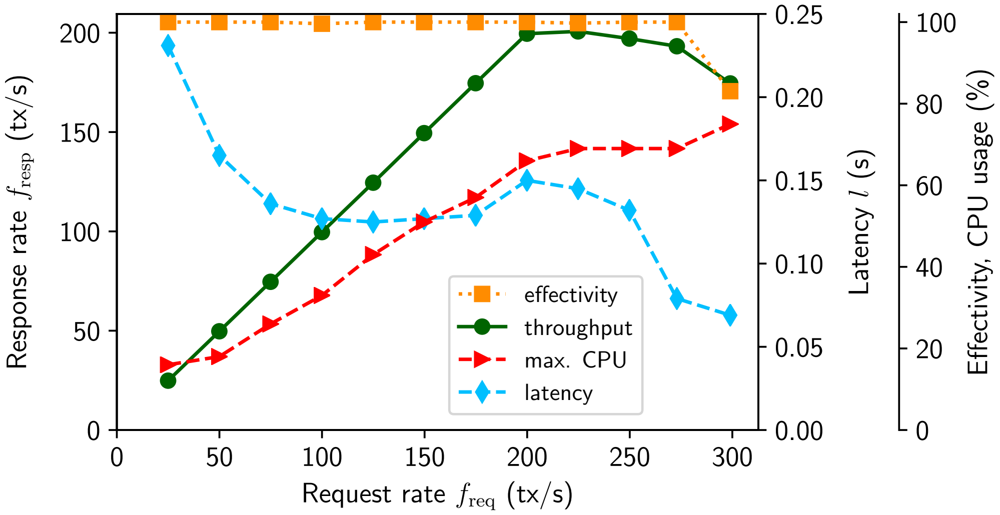


Figure 2: Ramping and localization logic.

$y_i = f_i \cdot x + t_i$ where $i \in \{\text{req}, \text{resp}\}$:

$$l(f_{\text{req}}) := \frac{t_{\text{req}}}{f_{\text{req}}} - \frac{t_{\text{resp}}}{f_{\text{resp}}} : \tag{5}$$

As long as the blockchain can handle the rate of requests, the two regressions will stay approximately parallel. $f_{\text{resp}}(f_{\text{req}})$ is monotonically increasing and close to the diagonal of the first quadrant. However, successively increasing f_{req}, at some point, the blockchain will not be able to keep up with the rate of requests anymore. Due to overload or congestion, we then expect that further increasing f_{req} will decrease f_{resp}. We can approximate $\hat{f}$ experimentally by successively increasing f_{req} until the regression slopes f_{req} and f_{resp} diverge, and, by (3), obtain an approximation for $\hat{f}$ by taking the maximum value for f_{resp} over all these trials. Fig. 2 shows one example for such a *ramping series* of measurements where we successively increase f_{req} by 25 tx/s. As expected, after a range where $f_{\text{resp}} \approx f_{\text{req}}$, throughput first stagnates and then declines. Further increasing f_{req} causes a drop in efficiency, the ratio of successfully operated transactions. Therefore, we can reasonably state that the maximum sustainable throughput $\hat{f}$ is approximately 200 tx/s in Figure 2.

To find the bottleneck responsible for the bound on throughput, we monitor the most relevant resource stats on nodes and clients. Here, for simplicity, we only discuss CPU usage on blockchain nodes. As expected, CPU usage is increasing in f_{req} and, therefore, with stress posed on the blockchain, it might well be the limiting resource in this case. On the other hand, the chart depicting latency seems surprising at first: One might assume that latency is also increasing in f_{req}. However, since responses are bundled in blocks, the creation of which is normally triggered by timeouts or reaching a certain number of pending transactions, it seems reasonable that for higher f_{req}, more blocks are produced per time and therefore the green staircase in Fig. 1 gets finer and moves closer to the request curve. Obviously, single transactions show a latency of only 0.1 s, and in Fig. 2, we can see that this is also close to the minimum overall latency at $f_{\mathrm{req}} \approx 150$. If we further increase f_{req}, the stress on the system dominates and – in line with intuition – latency increases until we reach $\hat{f}_{\mathrm{req}} \approx 200$. For $f_{\mathrm{req}} \geq 200$, we then see an unexpected drop in latency. However, we relativize this since, due to congestion and growing instability resulting from significant overload, the assumption that the response curve is close to a straight line turns out to be wrong. Hence, the derivation of latency is no more meaningful in this regime.

The heuristics and observations described above suggest an intuitive algorithm to efficiently determine $\hat{f}$ and $\hat{l}$ for a given DLT system. We have implemented this algorithm in the DLPS and display a simplification of the flowchart in Fig. 3. Before starting our experiment, we define parameters that are fixed throughout the whole benchmarking process, such as the duration of a single measurement period. We also specify an initial request frequency base and a step (25 tx/s in Figure 2) by which we increase f_{req} whenever the blockchain kept pace in the last trial. We have multiple criteria based on which we can decide whether or not a blockchain kept pace. The most crucial one ensures that f_{resp} may not significantly deviate from f_{req}:

$$\left|\frac{f_{\mathrm{resp}}}{f_{\mathrm{req}}} - 1\right| \leq \delta, \qquad \text{for reasonably small } \delta. \tag{6}$$

Moreover, we ensure that the response curve is actually close to a straight line by requiring that the coefficient of determination for the response curve is close to 1. To account for fluctuations in the system, we repeat a single trial multiple times if the blockchain cannot keep up according to our decision rule, because we do not want pure coincidence to stop a series as depicted in Figure 2 as long as we have not reached $\hat{f}$. We also require that a few (e.g., more than 3) successive increases of f_{req} have happened during the ramping series because since we want to measure maximum *sustainable* throughput, we also have to rule out that by coincidence, the system managed to deal with a very high rate for the duration.

Finally, when we have completed a ramping series, we run another ramping series with base and step suitably adjusted. We distinguish *localization runs*, in which we choose base

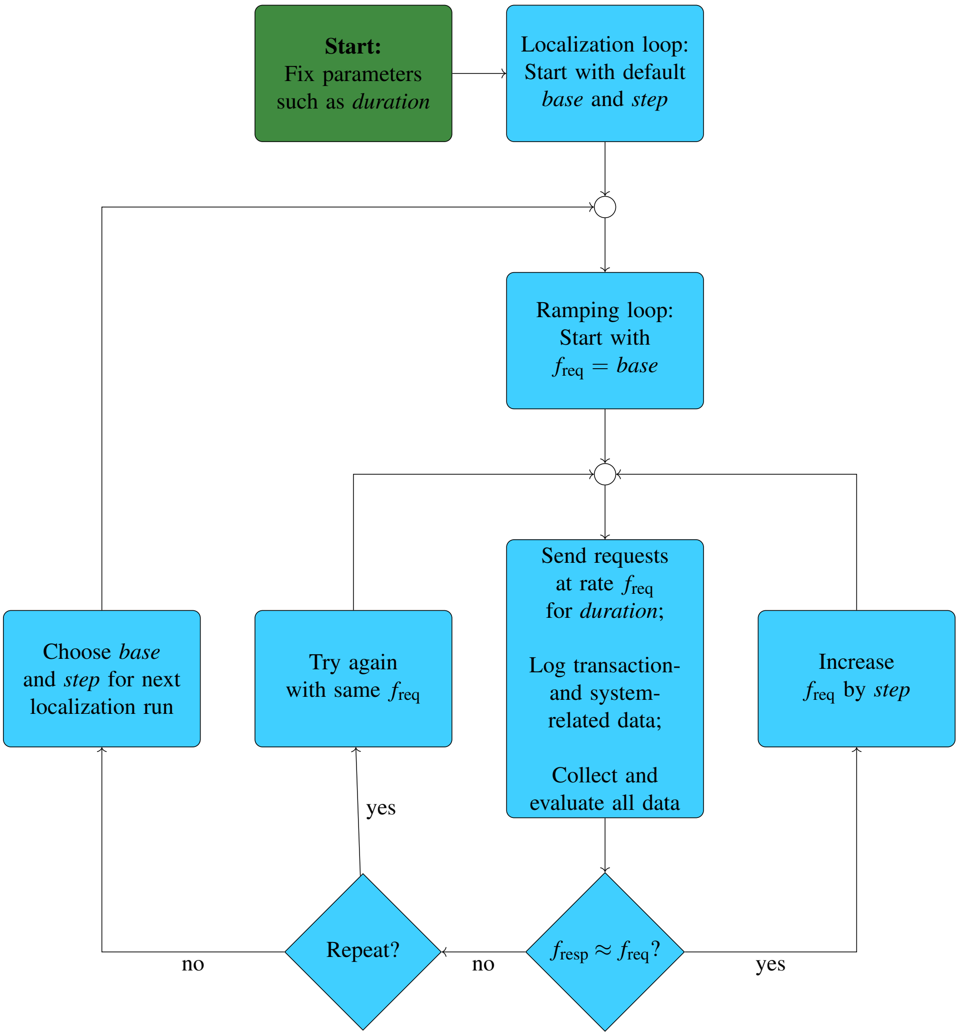

Start:
Fix parameters such as *duration*
Localization loop: Start with default *base* and *step*
Ramping loop: Start with $f_{\mathrm{req}} = base$
Send requests at rate f_{req} for *duration*;
Log transaction- and system-related data;
Collect and evaluate all data
Choose *base* and *step* for next localization run
Try again with same f_{req}
Increase f_{req} by *step*
Repeat?
$f_{\mathrm{resp}} \approx f_{\mathrm{req}}$?
yes
no
no
yes

Figure 3: The benchmarking process flowchart.

and step to get a higher resolution in the region around $\hat{f}_{\mathrm{req}}$ from the last ramping series (and significantly smaller values in kind the last series failed), and *repetition runs* in which we use the last value for base and step from a series of localization runs multiple times to obtain a statistically valid result. During all measurements, the DLPS uses established software for monitoring resources on both nodes and clients such as overall and single-core CPU usage (mpstat), memory (vmstat), disk utilization (iostat), network latencies (ping), and network traffic (ifstat).

D.4.2 Technical architecture

The DLPS framework consists of three Python packages to coordinate nodes and clients, trigger benchmarking functionalities, and aggregate and structure corresponding data. Fig. 4 shows the technical architecture of the DLPS.

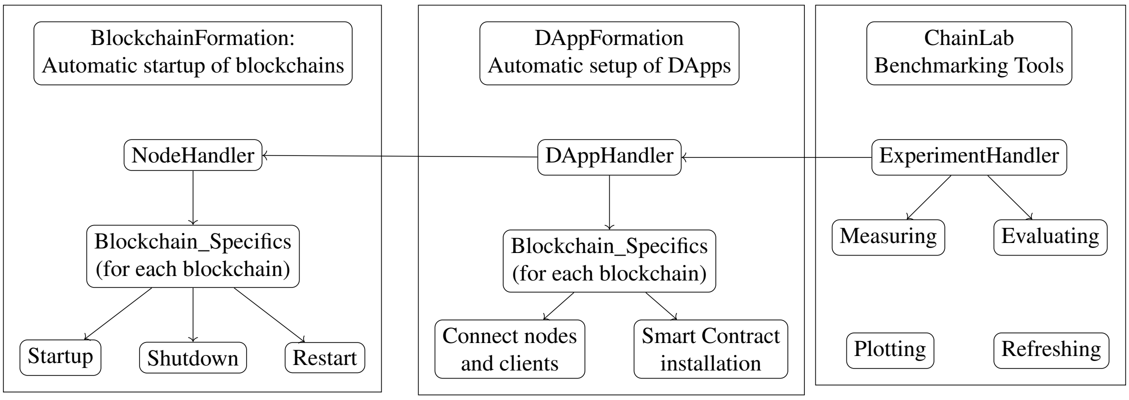


Figure 4: Architecture of the DLPS.

The package BlockchainFormation contains configurable and fully automatic startup-, restart-, and shutdown scripts for different permissioned blockchains. Since we want to offer highly customizable benchmarks, our NodeHandler launches and connects to virtual machines in Amazon Web Services (AWS) such that we can create and benchmark DLT networks of arbitrary size. However, the functionality could easily be extended to other cloud service providers or computing clusters, since the DLPS only relies on running ssh and scp-commands on the nodes. We have already implemented private Ethereum (with Geth and Parity client), Fabric, Indy, Quorum, and Sawtooth. Our NodeHandler integrates all these specifics and orchestrates the network startup.

The second repository, DAppFormation, consists of the required client-side functionalities. They serve to implement the blockchain-specific parts of the client setup, namely connecting client and blockchain networks. Moreover, for every blockchain, the associated clients wrap SC requests such that we can trigger the sending of requests at a specific rate f_{req} with a single script which is independent of the underlying blockchain.

All immediate functionalities for performance evaluation, in turn, are then integrated in ChainLab. Due to our wrappers in DAppFormation, we can implement the benchmarking logic (see Fig. 3) as well as the evaluation of our measurements in a blockchain-agnostic way. This design makes our benchmarking framework applicable to general DLT with a client-node architecture. Integrating another blockchain merely requires setting up startup scripts for both nodes and clients as well as the wrapper which serves to make the SC method calls from the benchmarking script blockchain-agnostic. The modular approach also allows for applying changes or extensions to the current frameworks at minimal effort and immediate impact on all tests within the framework. Finally, to make the data tidy and accessible, we stick to Wickham (2014) and provide a method to aggregate all measurements and their corresponding setup parameters into a single CSV file.

D.5 Performance characterization

D.5.1 Experimental setup

To illustrate the universal applicability of the DLPSs, we applied our framework to ten different network architectures for the five DLTs presented in Sec. D.2. We investigated network sizes of 1, 2, 4, 8, 16, 32, and 64 nodes, and used 32 clients distributed equally among the nodes in each setup. As workload, we chose a simple fundamental functionality on DLs, namely writing a single key-value pair into the ledger. This is the writeData basic workload mentioned in Sec. D.3, with a key space of size 10^4 and a value space of size 10^7.

We generally used cloud instances from the AWS m5 series because they balance CPU, memory, and network capabilities, all of which we consider relevant for a DLTs node. For the nodes, we decided to use m5.2xlarge instances in AWS in all experiments, which have 8 vCPUs and 16 GiB of RAM. For the configuration of the generic benchmarking process flow, we generally specified the following settings (see Sec. D.4). However, in some cases we had to make minor modifications to account for particularities of the setup, but these adaptions do not bring any bias to the results to the best knowledge of the authors.

- The duration of a measurement with fixed f_{req} was 20 s.
- To decide whether $f_{\mathrm{req}} \approx f_{\mathrm{resp}}$, we chose $\delta = 0.05$ in (6). We also specified a minimum coefficient of determination R^2 of 0.98, except for cases (generally, low throughput or large blocks) where the staircase was naturally very coarse. Generally, an R^2 value below our threshold occurred only rarely.
- We allowed for two retries in case $f_{\mathrm{resp}} \not\approx f_{\mathrm{req}}$ before breaking the ramping loop, and three consecutive rampings were required for a valid ramping series.
- In the localization runs, we used a success base rate of 80 % of the last ramping series' maximum throughput, with a step size of 4 %, and 50 % of resp. 4 % in case of a failure.
- We conducted three localization runs, followed by three repetition runs (see Sec. D.4).

All remaining parameters which completely determine the benchmarking process are included in the config files for the benchmarks and available in the DLPSs repository, among

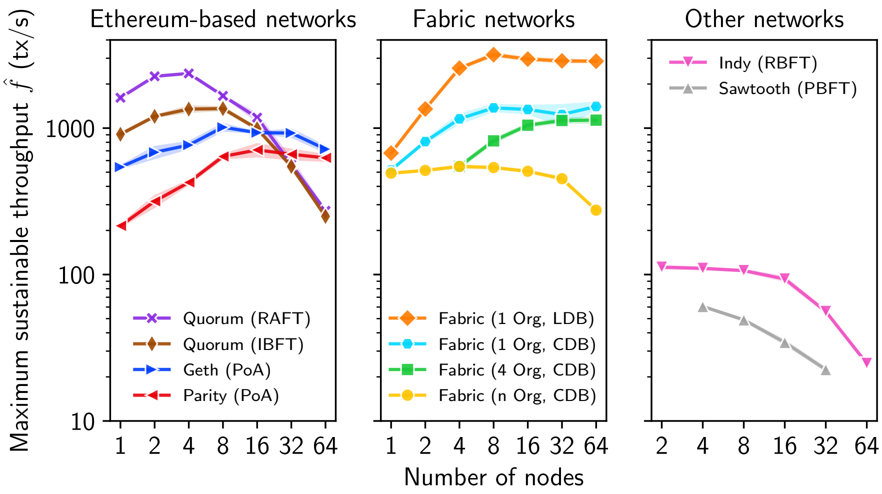


Figure 5: Results of our scalability analysis for different DLTs networks.

the results associated with the measurements presented here. Thereby, we address one of the key issues that led to the implementation of the DLPSs: The benchmarking process is transparent, and by repeating the experiments with the provided configuration files, our results are completely reproducible.

D.5.2 Experiment results

With the results from the experiments presented in Sec. D.5.1, we can give a thorough scalability analysis for these DLTs from a single standardized tool. Also, the range of investigated network sizes and in particular networks with 64 nodes are – to the best of our knowledge – unique in existing literature. Fig. 5 depicts our results for maximum sustainable throughput $\hat{f}$. Only the results of the three repetition runs went into the evaluation of each experiment. Each data point is the mean of these three results, with the shaded area specifying standard deviation.

We benchmarked private Ethereum with Geth and Parity PoA consensus and Quorum with IBFT and RAFT consensus. For Geth, Parity, and IBFT, we selected the minimum possible block time of 1 s, and for RAFT the default block time of 50 ms. Overall, Quorum with RAFT and IBFT consensus perform best for small networks, while Geth and Parity scale better for larger networks. The highest maximum sustainable throughputs $\hat{f}$ are 2 363±4 tx/s for 4-node Quorum with RAFT consensus and 1 350±60 tx/s for 4-node Quorum with IBFT consensus. For Geth, we observe maximum $\hat{f}$ of 1 010±50 tx/s at 8 nodes, while Parity reaches its maximum at 16 nodes with 710±70 tx/s. For all network sizes, RAFT has a higher $\hat{f}$ than IBFT, and Geth has higher $\hat{f}$ than Parity. The latter result makes sense because Aura consensus is more complex than Clique consensus in Geth. Our results on throughput for Parity are significantly higher than these of Dinh et al. (2017), who obtain a peak throughput of 46 tx/s and a latency of 3 s for Parity on an 8-node 8-client Parity network using Blockbench. On the other hand, Baliga et al. (2018)'s results on Quorum using Caliper are similar to our findings with the DLPS: Although they do not reach $\hat{f}$ because they had a limited number of clients and therefore bounded f_{req}, their evaluation suggests that throughput with RAFT is over 2 000 tx/s, and for IBFT around 2 000 tx/s. They also use hardware similar to the m5.2xlarge instances in our experiments, however, their cores had 3.6 GHz while the m5 instances only allow up to 3.1 GHz, indicating that the differences might stem from the experimental setup.

As described in Sec. D.2, Fabric separates the consensus layer (ordering service) from the rest of the system and offers customizable consensus options by specifying a SC's endorsement policy. Therefore, no canonical n-node setup exists to compare to the other DLTs benchmarked. Consequently, we defined different architectures[3]: (a) 1-Org, where we impose a trivial endorsement policy meaning that only a single node has to simulate the outcome of a transaction, and the SOLO orderer, (b) 4-Org, where, according to our endorsement policy, at least 4 nodes have to simulate and sign each transaction, and a separate 4-node RAFT network as ordering service, (c) n-Org, where all nodes need to endorse every transaction, and the ordering service is an equally sized RAFT network. In our opinion, the 4-Org case might be quite close to what one would implement in production, whereas 1-Org is the ideal case for performance but not really distributed, and n-Org is a worst case for performance and maybe the closest to the other networks in terms of consensus. We generally used CouchDB (CDB) as database because its support for complex queries makes it very suitable for enterprise DLTs solutions. To compare it with LevelDB (LDB), we investigated the 1-Org setup in both cases. In the 1-Org settings, we disabled encrypted messaging among the network (TLS), while enabling it in the 4-Org and n-Org case. Our first observation is that throughput of the 1-Org setup with LDB, which peaks at 3 180±80 tx/s for n=8, is approximately twice the performance of the 1-Org setup with CDB for large n, which reaches a maximum of $\hat{f} = 1\,410{\pm}90$ tx/s for n=64. Spot checks with better hardware and LDB also showed that our results are compatible with the ones of Androulaki et al., who measured more than 3 000 tx/s in

[3] Org is short for organizations: Due to the execute-order-validate paradigm, there are not only nodes (called peers in Fabric), but also clients and orderers that have an important role. It is intuitive to think about collections of these as orgs, e.g., one org might run an orderer, 4 peers and 8 clients, then they can trust a transaction if only one of their peers endorsed it, and contribute their own orderer to the CFT ordering service

these setups with a smart contract similarly complex as the writeData we used. Moreover, for both 1-Org setups and the 4-Org setup, we see that the overall shape of the curve seems like a saturation curve. Indeed, the ordering service does not change when we increase the number of nodes, but the number of potential endorsers increases. Hence, the endorsing tasks are split among more workers. On the other hand, the maximum download and validation rate for a node, which receives the blocks from the ordering service, poses an upper bound on throughput, which is independent on n as long as the ordering service itself is not the bottleneck. In contrast, we do not observe that saturation-like behaviour for the n-Org setup since here, the total endorsement workload increases like the number of nodes. This also explains why for small n, $\hat{f}$ is quite stable.

As highlighted in Sec. D.2, Indy does not support arbitrary SCs, so we could not define a writeData workload as straightforward as for the other frameworks. We decided to define issuance of a credential schema consisting of a single, random-number key as writeData transaction because this is a very simple on-chain write operation. For Indy and Sawtooth, we observe that $\hat{f}(n)$ is generally decreasing, which meets expectations for three-round consensus in RBFT resp. PBFTs. Indy shows almost constant $\hat{f}$ for n $\leq$ 16, with a maximum of $\hat{f} = 112 \pm 2$ tx/s for n=2. For n $\geq$16, $\hat{f}$ approximately halves for each subsequent doubling of the number of nodes, suggesting that our results truly display the overhead of BFTs-like consensus. The latency of Indy is around 3 s. We found no other benchmarks on Indy to compare with in our literature review and also when explicitly searching for performance results on Indy.

Although we also integrated Sawtooth with PoET and RAFT consensus in the DLPSs, we only systematically benchmarked Sawtooth with PBFTs consensus as RAFT setup turned out to crash frequently, and spot checking experiments for PoET consensus suggested that $\hat{f}$ is well below 30 tx/s in this case. PBFTs consensus in Sawtooth requires at least 4 nodes, and for n=64, we could not manage to set up a network that was running stable. We obtained a maximum of only $\hat{f}$=60.5±0.5 tx/s for n=4, and $\hat{l}$ grows from around 1.6 s for n=4 to 3 s for n=32. We noticed considerable performance improvements alongside the update to version 1.2 in October 2019: With version 1.0, we had never observed more than 8 tx/s, which is close to what Shi et al. (2019) measured for version 1.0. Moreover, frequent crashes had made systematic benchmarks almost impossible with this version.

D.6 Conclusion and future work

In this paper, we highlighted the current need for a transparent and universal framework to characterize the performance of DLT. To address this, we designed and implemented the DLPSs for determining key metrics and benchmarking applications end-to-end. The framework enables the creation of well-defined benchmarks (see Gray (1992)) due to four reasons. It is *transparent*, because we give clear definitions of latency and throughput as well as a description and implementation of the algorithm for measuring these. We offer *configurability* by supporting the automatic benchmarking of highly customizable architectures and parametrizing the benchmarking algorithm. By publishing the results of our measurements as well as our source code, one can easily reproduce the results (*repeatability*). Finally, the DLPSs is *extensible* as its modular implementation (see Fig. 4) allows the addition of new DLTs as well as adaptions of the benchmarking logic with reasonable effort. To demonstrate the applicability of the DLPSs to a large subset of DLTs, we conducted an in-depth study of performance and scalability properties of ten architectures for five permissioned DLTs on a broad range of network sizes (cf. Fig. 5). We plan to utilize the extensibility of our framework to develop the DLPSs further, including the integration of additional DLTs, such as Corda, and maintaining support for updates on the DLTs that we have already included. We will further extend the parameters that one can choose, and provide further tools for evaluating the gathered data, hence further reducing the hurdle to establishing the DLPSs as a standard tool to measure DLTs performance.

References

Androulaki, E. et al. (2018). “Hyperledger Fabric: A distributed operating system for permissioned blockchains”. In: *Proceedings of the 13th EuroSys Conference*. DOI: 10.1145/3190508.3190538.

Angelis, S. de, L. Aniello, F. Lombardi, A. Margheri, and V. Sassone (2017). *PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain*. URL: https://eprints.soton.ac.uk/415083/2/itasec18_main.pdf.

Aublin, P.-L., S. B. Mokhtar, and V. Quéma (2013). “RBFT: Redundant Byzantine fault tolerance”. In: *33rd International Conference on Distributed Computing Systems*. IEEE, pp. 297–306. DOI: 10.1109/icdcs.2013.53.

Baliga, A., I. Subhod, P. Kamat, and S. Chatterjee (2018). *Performance evaluation of the Quorum blockchain platform*. URL: http://arxiv.org/abs/1809.03421.

Buterin, V. et al. (2014). *A next-generation smart contract and decentralized application platform*. URL: https://github.com/ethereum/wiki/wiki/White-Paper.

Butijn, B.-J., D. A. Tamburri, and W.-J. van den Heuvel (2020). “Blockchains: A systematic multivocal literature review”. In: *ACM Computing Surveys* 53 (3). DOI: 10.1145/3369052.

Castro, M., B. Liskov, et al. (1999). “Practical Byzantine fault tolerance”. In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation*, pp. 173–186. URL: https://pmg.csail.mit.edu/papers/osdi99.pdf.

Christidis, K. and M. Devetsikiotis (2016). “Blockchains and smart contracts for the Internet of Things”. In: *IEEE Access* 4. DOI: 10.1109/access.2016.2566339.

Dinh, T. T. A., J. Wang, G. Chen, R. Liu, B. C. Ooi, and K. Tan (2017). *Blockbench: A framework for analyzing private blockchains*. URL: http://arxiv.org/abs/1703.04057.

Fan, C., S. Ghaemi, H. Khazaei, and P. Musilek (2020). “Performance evaluation of blockchain systems: A systematic survey”. In: *IEEE Access* 8, pp. 126927–126950. DOI: 10.1109/access.2020.3006078.

Gervais, A., G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun (2016). “On the security and performance of proof of work blockchains”. In: *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security*, pp. 3–16. DOI: 10.1145/2976749.2978341.

Gray, J. (1992). *Benchmark handbook: for database and transaction processing systems*. Morgan Kaufmann Publishers Inc.

Hyperledger Fabric Repository (2022). URL: https://github.com/hyperledger/fabric.

Hyperledger Indy repository (2020). URL: https://github.com/hyperledger/indy-node.

Iansiti, M. and K. R. Lakhani (2017). “The truth about blockchain”. In: *Harvard Business Review* 95 (1), pp. 118–127. URL: https://hbr.org/2017/01/the-truth-about-blockchain.

Jain, R. (1991). *The art of computer systems performance analysis – techniques for experimental design, measurement, simulation, and modeling*. Wiley.

Jensen, T., J. Hedman, and S. Henningsson (2019). “How TradeLens delivers business value with blockchain technology”. In: *MIS Quarterly Executive* 18 (4), pp. 221–243. DOI: 10.17705/2msqe.00018.

Lamport, L., R. Shostak, and M. Pease (1982). “The Byzantine generals problem”. In: *ACM Transactions on Programming Languages and Systems* 4 (3), pp. 382–401. DOI: 10.1145/3335772.3335936.

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

Ongaro, D. and J. Ousterhout (2014). “In search of an understandable consensus algorithm”. In: *USENIX Annual Technical Conference*, pp. 305–319. URL: https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf.

Ousterhout, J. (2018). “Always measure one level deeper”. In: *Communications of the ACM* 61 (7), pp. 74–83. DOI: 10.1145/3213770.

Poon, J. and T. Dryja (2016). *The Bitcoin Lightning network: Scalable off-chain instant payments*. URL: https://www.bitcoinlightning.com/wp-content/uploads/2018/03/lightning-network-paper.pdf.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). “Building a blockchain application that complies with the EU general data protection regulation”. In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Saltini, R. (2019). *Correctness analysis of IBFT*. https://arxiv.org/pdf/1901.07160.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020). “The energy consumption of blockchain technology: Beyond myth”. In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

Shi, Z., H. Zhou, Y. Hu, S. Jayachander, C. de Laat, and Z. Zhao (2019). “Operating permissioned blockchain in clouds: A performance study of Hyperledger Sawtooth”. In: *18th International Symposium on Parallel and Distributed Computing*. IEEE, pp. 50–57. DOI: 10.1109/ispdc.2019.00010.

Szabo, N. (1997). “Formalizing and securing relationships on public networks”. In: *First Monday* 2 (9). DOI: 10.5210/fm.v2i9.548.

Vukolić, M. (2016). “The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication”. In: *Proceedings of the International Workshop on Open Problems in Network Security*. Springer, pp. 112–125. DOI: 10.1007/978-3-319-39028-4_9.

Wickham, H. (2014). “Tidy data”. In: *Journal of Statistical Software* 59 (10). DOI: 10.18637/jss.v059.i10.

E Research Paper 4 – An in-depth investigation of the performance characteristics of Hyperledger Fabric

Authors:

Tobias Guggenberger, Johannes Sedlmeir, Gilbert Fridgen, & André Luckow

Published in:

Computers & Industrial Engineering (DOI: 10.1016/j.cie.2022.108716)

Abstract:

Private permissioned blockchains are deployed in ever greater numbers to facilitate cross-organizational processes in various industries, particularly in supply chain management. One popular example of this trend is Hyperledger Fabric. Compared to public permissionless blockchains, it promises improved performance and provides certain features that address key requirements of enterprises. However, also permissioned blockchains are still not as scalable as centralized systems, and due to the scarcity of theoretical results and empirical data, their real-world performance cannot be predicted with the necessary precision. We intend to address this issue by conducting an in-depth performance analysis of Hyperledger Fabric. The paper presents a detailed compilation of various performance characteristics using an enhanced version of the Distributed Ledger Performance Scan (DLPS). Researchers and practitioners alike can use the various performance properties identified and discussed as guidelines to better configure and implement their Hyperledger Fabric network. Likewise, they are encouraged to use the DLPS framework to conduct their measurements.

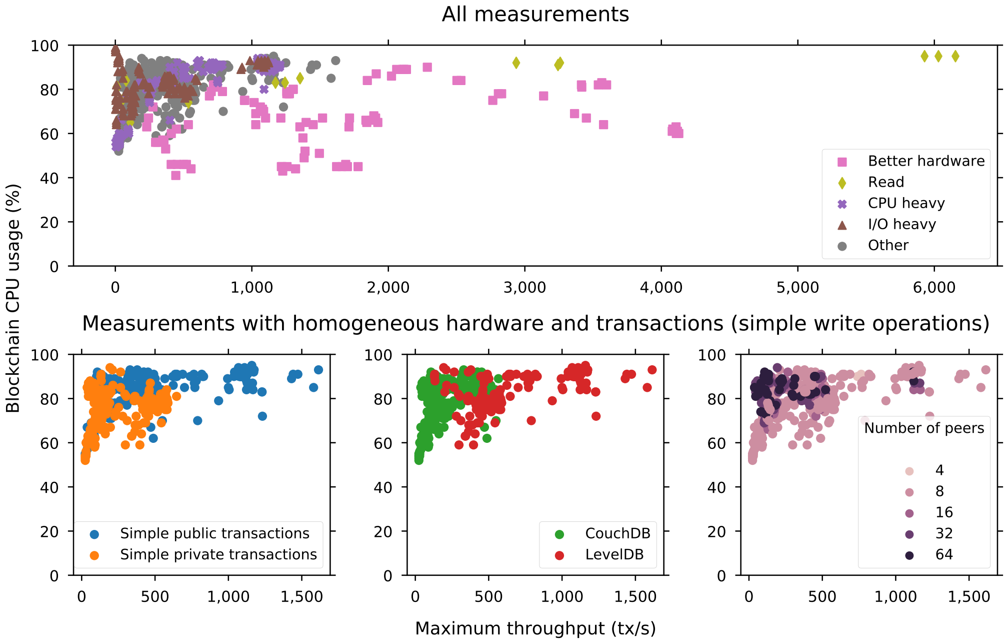
Keywords:

Benchmarking, blockchain, distributed ledger, scalability, supply chain, throughput

Highlights

- Performance is a widely acknowledged challenge for industrial blockchains.
- Numerous parameters impact Fabric's maximum throughput; latency is consistently low.
- Hardware, database, network size, and privacy can have a high impact on throughput.
- Fabric offers good performance even in large intercontinental supply chain networks.

Graphical Abstract



This chart illustrates influential experimental parameters on the throughput of Hyperledger Fabric networks. In this paper, we illustrate that multiple parameters have a significant impact on performance metrics. Our results were collected by setting up more than 1,500 Hyperledger Fabric networks and operating more than 200 million transactions in experiments that ran for more than 2,000 hours. The purpose of these efforts is to validate and extend previous research by evaluating more than 15 network- and transaction-related parameters, including choices of hardware and database, transaction payloads and privacy configurations, different network sizes of 5 to 128 nodes, and geographic distribution. We also analyze the impact of node crashes.

E.1 Introduction

Bitcoin was the first blockchain system, developed primarily for the decentralization of financial systems. It created a new virtual currency that allows transactions without the involvement of distinct intermediaries like banks (Nakamoto, 2008). Based on this general concept, Buterin et al. (2014) extended the scope of blockchain technology to a broader field of application. At that time, blockchain technology was only able to provide users with limited programming logic, so Ethereum improved its versatility by introducing a programming language and an associated runtime environment ("Ethereum virtual machine") to execute smart contracts. These were first conceptualized by Szabo (1997), and even in their early form, they facilitated the execution of highly customizable program code in a peer-to-peer (P2P) environment without relying on a distinct intermediary. The advancement of blockchain technology has fostered the development of decentralized applications in the business and indeed in the public sector, while they have gone far beyond the initial use cases in the financial sector (Casino et al., 2019; Labazova et al., 2019) for cross-organizational workflows (Fridgen et al., 2018). This progress extends to applications in the public sector (Rieger et al., 2019), the pharmaceutical sector (Mattke et al., 2019), and the automotive sector (Miehle et al., 2019). Especially within supply chain management, many researchers agree that blockchain provides a viable infrastructure that facilitates a more efficient way of sharing information, improved data quality, and traceable product provenance (Agrawal et al., 2021; Azzi et al., 2019; Guggenberger et al., 2020; Jensen et al., 2019; Lim et al., 2021; Longo et al., 2019; Reddy et al., 2021; Sunny et al., 2020).

In their comprehensive literature review of current developments and potential applications of blockchain in supply chain management, Chang and Chen (2020) conclude that blockchain has the potential to disrupt supply chain operations and provide not only distributed governance and process automation but also improved performance across the board. Yet despite the considerable benefits that distributed ledgers can offer enterprises, such as consolidating audit and production data in an unimpeachable distributed database, public blockchains are still subject to numerous limitations. Examples include their generally rather high transaction fees, their lack of finality, and their inability to safeguard transaction confidentiality (Kannengießer et al., 2020; Sedlmeir et al., 2022b). Many permissionless blockchains are also based on Proof of Work, which is why they are extremely high in energy consumption (Sedlmeir et al., 2020); an inconvenient truth that is difficult to reconcile with corporate sustainability goals. Fortunately, we were able to note a broad awareness of the challenges concerning throughput and latency. In conducting a systematic study of the literature on blockchain-based supply chain applications, we found that 83 of 128 publications acknowledged performance challenges. Meanwhile, privacy issues were noted by 71, security issues by 60, regulatory issues by 47, and challenges concerning costs by 41. To name but one example, Perboli et al. (2018) analyzed a supply chain use case and mentioned multiple times that performance evaluation is a crucial step in the design and implementation of blockchain-based supply chain solutions.

In attempts to resolve these issues and answer the increasing demand for enterprise-level blockchain applications expressed in various industries, developers have introduced new frameworks and modified blockchain architectures. These are intended to compensate for the shortcomings of public permissionless blockchains and adapt them to the needs of enterprises (Kannengießer et al., 2020). To achieve these goals, frameworks were developed in such a way as to implement private permissioned blockchains that restrict participation in the blockchain and consensus to a consortium (Beck et al., 2018). While other blockchains like Hyperledger Sawtooth have also been examined with regard to their potential to support supply chain applications (Perboli et al., 2020), Hyperledger Fabric (Fabric) has arguably become the preeminent technical support structure in this domain. Indeed, Guggenberger et al. (2020), Lim et al. (2021), and Reddy et al. (2021) have all discussed the use of Fabric for large-scale cross-enterprise applications, mainly because the framework provides high security and performance as well as flexible tools for managing access, safeguarding privacy, and implementing business logic (Androulaki et al., 2018; Kannengießer et al., 2020).

At present, several major projects based on Fabric are transitioning from tests and minimum viable products with limited scope to production-ready systems, as a result of which there is a growing number of participating parties and operations in these projects (IBM, 2020a; Miehle et al., 2019). When looking at these projects, however, it becomes apparent that the requirements concerning private or public transactions, the varying complexities of smart contracts, and the need to support and adapt network topologies all differ significantly (Kannengießer et al., 2020). It is a valuable asset, therefore, that Fabric offers various configurations which allow one to adapt it to a wide range of different use case requirements (Kolb et al., 2020). This is especially significant in projects like TradeLens (Jensen et al., 2019) since its purpose is to provide the infrastructure for worldwide supply chain monitoring, which means that it places extensive requirements on the performance of blockchain systems. The choice of various architectural parameters, such as network size, hardware configuration, internet connection speed, and complexity of operations (i.e., smart contracts methods), is known to have a large impact on the performance of distributed systems in general and in particular on that of blockchains (see, e.g., Baliga et al., 2018b; Thakkar et al., 2018). It is inevitable, then, that trade-offs between security, network size, privacy, and performance must be considered when designing a system with high performance and reliability requirements (Kannengießer et al., 2020).

In Section E.3 of this paper, our literature review identifies two significant gaps in the current body of knowledge on the general performance of permissioned blockchains and the specific performance of Fabric. The first of these gaps is the result of the fact that, to date, studies have focused on particular variables without allowing for a holistic view, mainly because these studies have conducted their measurements with non-standardized tools. Furthermore, many have allowed for ambiguity in the definition of their key metrics and indeed in the attainment of their results. Therefore, multiple observations are neither replicable nor generalizable (Sedlmeir et al., 2021), and until they are, there can be no holistic view of the performance of Fabric. As for the second research gap, this is the result of the fact that Fabric has been developing at a considerable pace, frequently offering new configuration options and features that impact its performance at a rate too fast and extensive to have been covered by the limited literature to date. For example, since private data collections can provide a certain level of access control in a cross-enterprise system, they are essential for many enterprise-level applications (Kolb et al., 2020; Sedlmeir et al., 2022a; Zhang et al., 2019). The private data transaction process, however, is far more complex than the conventional transaction process because it introduces additional gossip routines. These protocol changes make it difficult to predict the performance of private data transactions compared to that of conventional transactions, and yet, to the best of our knowledge, there has been no rigorous academic study of the many ways in which the use of private data collections can impact performance.

In this paper, we intend to close these research gaps by studying a wide variety of Fabric's performance characteristics. Our in-depth analysis thereof comprises the perspectives of both researchers and architects of large-scale enterprise and public sector projects. Our measurements significantly extend the range of performance characteristics studied to date, including additional scenarios that are highly relevant to the real-world use of blockchain technology, e.g., in supply chain applications in the industrial and public sectors. As Kannengießer et al. (2020) have pointed out, the right balance of these factors is essential if one's deployment of blockchain is to facilitate the most effective creation of value. Our research objective, therefore, is to develop a list of relevant variables, measure their specific impact on different Fabric implementations, and demonstrate the potential of Fabric in various scenarios. In doing so, we hope to advance the understanding of enterprise blockchains and what they represent; a highly complex fault-tolerant distributed system applied in real-world settings, as required by enterprises. To be more specific, we put the spotlight on the capabilities of Fabric, one of the blockchain frameworks most frequently used by industry consortia. While Fabric has been discussed by Lim et al. (2021) and Reddy et al. (2021) as an infrastructure for supply chain systems supported by blockchain, we provide further insights into how private permissioned blockchains can support large-scale and indeed global supply chains. Our findings include that Fabric scales exceedingly well with CPU-heavy transactions but struggles with transaction payloads larger than 100 kB. Furthermore, whereas Fabric is very suitable for intercontinental networks, private transactions, in particular, suffer from commensurate high latency. Since our overarching research aim is to close many of the gaps in the knowledge on the performance characteristics of blockchain, this paper also provides an extension of the distributed ledger performance scan (DLPS) (Sedlmeir et al., 2021), a blockchain benchmarking framework. The DLPS offers not only clear definitions of key performance metrics but also an end-to-end description of their setup and measurement, which ensures full transparency and repeatability. We supply our extension of the DLPS in the open-source repository (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021) along with the results of our experiments so that researchers can repeat our measurements or use them to easily examine new configurations.

The remainder of this paper is structured as follows: Section E.2 gives an overview of the key concepts on which Fabric and its architecture are predicated. Section E.3 provides a thorough review of the literature on benchmarking Fabric and identifies the main gaps to be closed. Section E.4 describes the measurement process involved in the DLPS in detail. Section E.5 then presents the main findings of this study by demonstrating our benchmarking results under careful consideration of the wide range of variables employed in the benchmark tests. In Section E.6, we discuss our findings, outline their implications for real-world applications, and provide design guidelines. Finally, in Section E.7, we identify opportunities for future research.

E.2 Hyperledger Fabric: Technical background

E.2.1 System architecture

Since the days of version 1.0, Fabric has facilitated a paradigm that fundamentally differs from most blockchains in that it offers improved performance, flexibility, and privacy

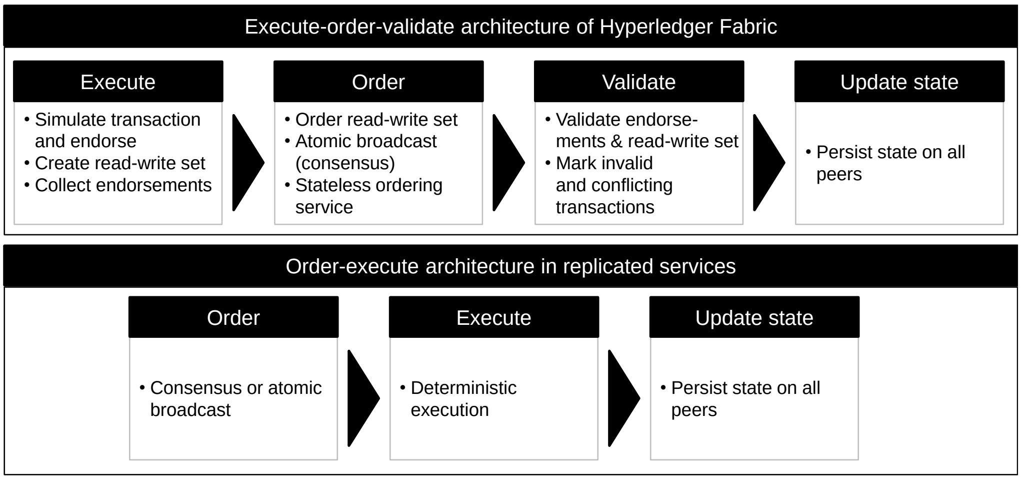

Execute-order-validate architecture of Hyperledger Fabric
Execute
• Simulate transaction and endorse
• Create read-write set
• Collect endorsements
Order
• Order read-write set
• Atomic broadcast (consensus)
• Stateless ordering service
Validate
• Validate endorsements & read-write set
• Mark invalid and conflicting transactions
Update state
• Persist state on all peers
Order-execute architecture in replicated services
Order
• Consensus or atomic broadcast
Execute
• Deterministic execution
Update state
• Persist state on all peers

Figure 1: The execute-order-validate paradigm in Fabric compared to the order-execute architecture common to most blockchains.

features (Kolb et al., 2020). Instead of relying on an order-execute architecture, Fabric uses an execute-order-validate paradigm (see Figure 1). Order-execute means two things: first, that the consensus mechanism is responsible for ordering and then broadcasting new transactions, and second, that all peers execute these transactions sequentially. In contrast, execute-order-validate implies that Fabric separates execution and validation from ordering (Androulaki et al., 2018).

This altered replication process requires a new system architecture. A Fabric node can perform one of the following three roles (Androulaki et al., 2018):

- Clients are responsible for submitting a transaction proposal to their peers and broadcasting any transactions in the form of a bundled endorsement response to bring transactions into order (Androulaki et al., 2018).

- Peers receive the transaction proposals from clients, simulate them, and send the signed result back to the clients. Eventually, they validate transactions. All peers maintain a ledger consisting of an append-only data structure (blockchain) of all previous transactions and a structure that represents the latest world state of the ledger. To store this ledger state, peers can use conventional databases. At present, Fabric 2.0 supports LevelDB and CouchDB. Due to the execute-order-validate paradigm, Fabric does not require all peers to execute all transaction proposals. This design feature sets Fabric apart from most other blockchains, be they public permissionless or private permissioned blockchains. By means of an endorsement policy, one can specify the subset of peers required for the transaction proposal execution, and this can be done individually for each smart contract method. These subsets of peers are also called endorsers or endorsing peers (Androulaki et al., 2018).

- Ordering Service Nodes, also known as orderers, together form the ordering service. This ordering service is responsible for creating the total order of all transactions. There are different ways of implementing the ordering service, ranging from a (now deprecated) solo orderer to distributed protocols, such as RAFT (Ongaro and Ousterhout, 2014) and Kafka (Kreps et al., 2011). While these protocols already address different levels of fault tolerance (Androulaki et al., 2018), developers are working on future ordering services that should also account for byzantine faults (Lamport et al., 1982).

Clients, peers, and orderers are further grouped into organizations (abbreviated as orgs). These typically represent companies or wider groups of participants. Based on their organizational affiliation, these entities have different rights, like the permission to join a blockchain channel that represents a private subnet of communication between two or more network participants including a corresponding ledger. A peer can either join one or multiple channels. Those who opt for a greater number of nodes in one organization will experience increased redundancy and, therefore, reduced efficiency and networking overhead. On the other hand, the distribution of the simulation workload to more servers approach facilitates parallelization and, thus, a higher throughput of endorsements of transactions (Thakkar and Natarajan, 2021).

E.2.2 Transaction flow

Fabric's execute-order-validate paradigm separates the transaction flow into three parts: i) execution (sometimes also referred to as simulation) of a transaction, which involves checking its correctness by comparing the signed result of redundant execution on different peers. This is also called an endorsement. ii) ordering, which is done by means of a consensus protocol, regardless of the semantics of a transaction. iii) transaction validation, which ensures the endorsement policy and state consistency (Androulaki et al., 2018). Figure 2 provides an overview of the transaction flow.

(i) **Execution phase:** A client sends a cryptographically signed transaction proposal to one or more endorsing peers for execution (simulation). The peers do not yet update their ledger but only generate a read set and a write set (1). The write set consists of all key updates resulting from the simulation, whereas the read set contains all keys

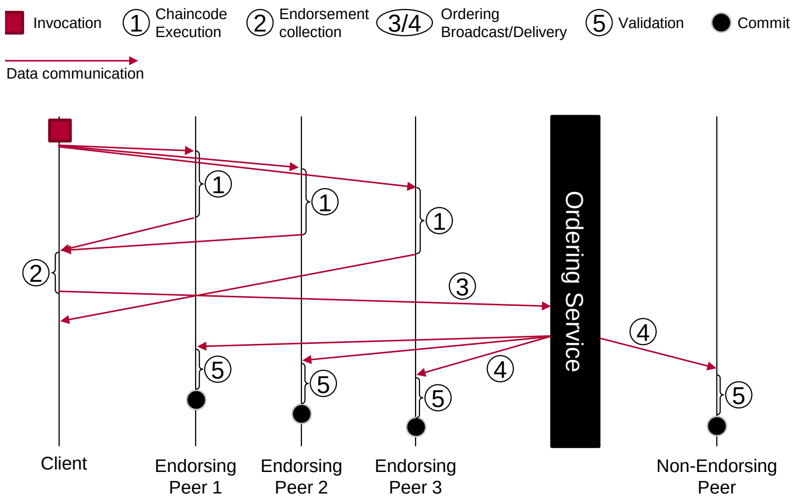


Figure 2: Fabric high-level transaction flow, adapted from Androulaki et al. (2018).

that the peers read during the simulation. The endorsers then create a cryptographically signed endorsement, including the read and write sets, and send this back to the client in the form of a proposal response. The client collects endorsements until the requirements set by the endorsement policy are met (2). This action further ensures that enough endorsers produce the same execution result and, in doing so, respond with the same read-write set (Androulaki et al., 2018).

(ii) **Ordering phase:** Once the client has received enough consistent endorsements, the client bundles them all, creates a signed transaction, and sends it to the ordering service (3). The ordering service uses consensus to establish a total order of all transactions. Furthermore, the ordering service batches the transactions in blocks and signs them (Androulaki et al., 2018).

(iii) **Validation phase:** Blocks can either be delivered directly by the ordering service or indirectly by other peers who do so through a gossip protocol (4). When a new block is delivered to a peer, it enters the validation phase (5), which involves the following three sequential steps (Androulaki et al., 2018):

a. The peer checks whether every transaction meets the endorsement requirements. If a transaction is invalid, the peer will mark it accordingly and ignore its effect.

b. The peer enters the ledger update ("commit") phase and appends the block to the local store ledger. For each transaction not marked as invalid, the peer writes all key-value pairs of the write set to the local state. Therefore, Fabric records

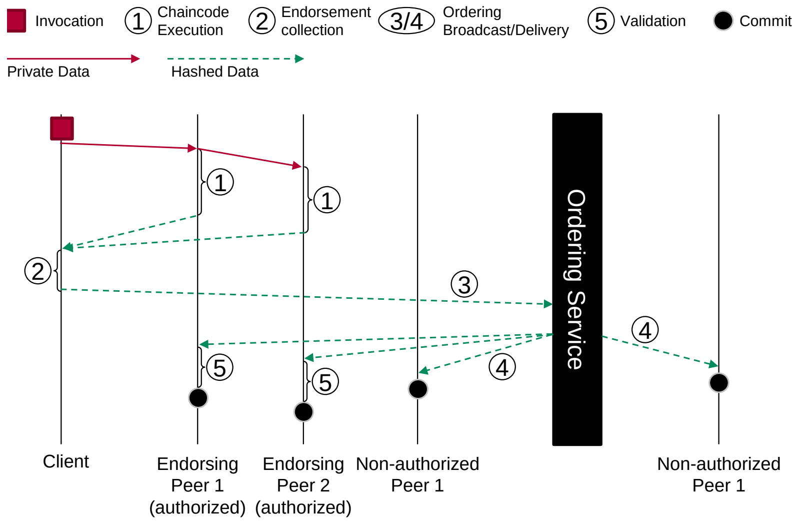


Figure 3: Fabric private data high-level transaction flow, adapted from Androulaki et al. (2018).

invalid transactions even though they do not affect the state (Androulaki et al., 2018).

E.2.3 Private data transaction flow

Since version 1.2, Fabric also supports private data by means of private data collections. These represent privacy policies that determine which peers ought to process and store related data and which organizations should be able to access it (Ma et al., 2019). Private data in particular benefits from the execute-order-validate paradigm and endorsement policies that do not require every peer to recompute every transaction in order to validate it. This feature makes it possible to conduct transactions where only a subset of the organizations participating on the Fabric blockchain store the actual data, while the remaining organizations only see the transaction hash, without relying on complex cryptographic techniques, such as zero-knowledge proofs or homomorphic encryption (IBM, 2020b).

Private data is mainly handled in accordance with the standard protocol discussed in Section E.2.2, but at certain stages it departs from that protocol to ensure confidentiality in the three phases of execution, ordering, and validation (see Figure 3).

(i) **Execution phase:**

The client sends a proposal request, including the confidential data, to the designated endorser of the authorized organizations. Based on the collection policy that defines which organizations should be able to access this private data, the endorsing peers distribute it to other authorized peers via a gossip protocol. All peers in receipt of the private data store it in a transient data store. Similar to the general transaction flow, the endorsers generate a read-write set and send this to the client in the form of an endorsement (1). These read-write sets do not contain any confidential data, however, but rather a hash of the private data keys and values. Once the client has received enough endorsements (2), the client will send a transaction to the ordering service (3) responsible for the total order of transactions (Ma et al., 2019).

(ii) **Ordering phase:** The ordering phase follows a similar sequence as the general transaction flow. By consensus, the orderers include the transactions in a block and distribute them to all peers (4). Therefore, all peers receive the hashes of the private data, which facilitates subsequent validation (Ma et al., 2019).

(iii) **Validation phase:** All peers will store the transaction in their ledger and update the read-write set with the associated hash values. Furthermore, in case a peer is authorized to access the private data related to the transaction, the peer will check the transient data store for the private data. If the peer has not received the private data in the execution phase, the peer will try to pull the private data from other authorized peers, whereupon the peer will use the hash values of the transaction to validate the private data and eventually save it to the private state database (Ma et al., 2019). Generally, the peer makes use of one more table than the regular state database.

While private data can be transferred confidentially between specific organizations, the required certificates are still used in plain text in order to verify permissions. Even though this is done without insight into the content of a transaction, this process entails severe confidentiality issues as it is apparent who is issuing new transactions. To safeguard against this confidentiality breach and hide the identity of the issuing client certificate, IBM introduced an implementation of the identity mixer protocol (Bichsel et al., 2009; Camenisch et al., 2017). However, not only is this feature highly experimental. To date, it is only supported in the Java-implementation of the Fabric client.

E.3 Related work

While the performance of blockchains is often considered crucial when working towards production-level systems (Thakkar and Natarajan, 2021), at the time of writing, research in the field of systematic benchmarking is still scarce. To gain a full understanding of what research there is on the performance of Fabric, we conducted a structured literature review. To ensure the inclusion of all relevant publications, we first defined "Hyperledger AND Fabric" as a search string. We then used the string for queries in ACM Digital Library, AIS electronic Library, arXiv, IEEE Explore, and Web of Science. The initial set of search results totaled 1085 papers. After an initial screening of their titles and abstracts, we excluded 1007 publications for lack of relevance. Based on a subsequent full-text analysis, we removed every paper that performed benchmarking on a heavily modified version of Fabric as their findings are highly theoretical and not transferable to the publicly available versions of the system. After those eliminations, we were left with 19 articles that analyze the performance and scalability of Fabric. Table 1 depicts this final set, the collective intelligence of which informed the following pages.

The study of Pongnumkul et al. (2017) marks the first thorough performance analysis of Fabric. By comparing the Go implementation of the Ethereum client (Geth) to Fabric, the authors demonstrated the potential performance benefits of using a private permissioned blockchain. The following year, Dinh et al. (2018) standardized a performance analysis of private permissioned blockchains by introducing the first systematic benchmarking framework: Blockbench (*Blockbench repository* 2022). Blockbench makes heavy use of Yahoo! Cloud Serving Benchmark and Smallbank, both of which are benchmarking frameworks for conventional IT systems with a focus on centralized databases. The authors also compared Fabric to Geth and Parity. Rather than opt for the re-architectured v1.0 of Fabric, Dinh et al. (2018) used v0.6 for their comparison as they gained far better performance results with the older release.

Subsequent work has made almost exclusive use of Fabric $\geq$v1.0. Compared to the findings of Dinh et al. (2018), who accomplished approximately 1,000 transactions per second, Androulaki et al. (2018) attained far higher performance statistics with the newly introduced architecture of v1.x. Having analyzed the preview version of v1.1 in detail, they demonstrated that Fabric has the potential to cope with over 100 peers and, in the right circumstances, perform more than 3,500 transactions per second. However, since the performance results of Baliga et al. (2018a) were significantly lower than those of Androulaki et al. (2018), it is clear that the potential performance of Fabric is contingent on various factors, such as its benchmarking framework, its release version, and the employed hardware. Therefore, later research extended testing to multiple parameters and newer release versions of Fabric. For instance, Thakkar and Natarajan (2021) and Kuzlu et al. (2019) analyzed v1.4 of Fabric, and both studies lend further credence to the complexity of performance tests of blockchain systems. In particular, Kuzlu et al. (2019)

Source	Detailed content
Pongnumkul et al. (2017).	This article presents a methodology for evaluating the performance of Ethereum and Fabric. The research team derives performance figures for execution time, latency, and throughput, while also considering various workloads.
Androulaki et al. (2018).	This paper presents the execute-order-validate blockchain architecture of Fabric v1.1.0. The research team examines the throughput and latency under consideration of various parameters, such as block size, number of vCPUs, and number of peers.
Baliga et al. (2018a).	This study makes use of Caliper to examine the performance of Fabric v1.0. The authors consider various influencing factors, including the number of nodes, endorsement policy, block size, and transaction size.
Dinh et al. (2018).	The authors of this paper present the first systematic benchmarking framework for permissioned blockchains: Blockbench. It builds on the established YCSB and Smallbank frameworks to allow benchmarking of private Ethereum (Geth, Parity), Fabric, and Quorum. With the use of this framework, the authors compare the performance of Fabric to Ethereum.
Hao et al. (2018).	This article presents a method of evaluating the performance of consensus algorithms in Ethereum and Fabric. The authors derive performance figures for latency and throughput, while also considering varying workloads.
Nasir et al. (2018).	The authors of this study compare the performances of Fabric v1.0 and v0.6. As well as analyzing execution time, latency, and throughput, they also vary the number of nodes to examine the scalability of the two implementations.
Thakkar et al. (2018).	This study examines the impact of various factors on Fabric v1.1, such as block size, endorsement policy, channels, and state database choice. The authors identify performance bottlenecks and propose optimizations subsequently included in later versions of Fabric.
Koushik et al. (2019).	With the help of the Caliper benchmarking framework, the authors of this article investigate the performance of Fabric with regard to transaction throughput, latency, and send rate. They also analyze the impact of varying numbers of organizations.
Kuzlu et al. (2019).	Again with the help of Caliper, this research team examines the performance of Fabric with regard to throughput, response time, and simultaneous transactions.
Nguyen et al. (2019).	The authors of this study use a customized version of the Hyperledger Caliper benchmarking framework to examine the impact of sup-second network delays on the performance of Fabric. To create the Fabric network, they set up their test with two cloud instances, one in Germany and one in France.
Dabbagh et al. (2020).	The authors of this study use the Caliper benchmarking framework to compare the performance of Fabric with that of Ethereum. They also evaluate different versions of the Fabric SDK.
Dreyer et al. (2020).	For this research project, the authors analyze the performance of Fabric by creating various network configurations and measuring throughput, latency, and error rate, along with the overall scalability of the Fabric platform. The results are presented in the context of older versions of Fabric.
Geneiatakis et al. (2020).	The authors focus on the application of blockchain in the field of cross-border e-government services. However, they also take separate account of the performance of Fabric. Among other variables, they discuss network delay as a key factor.
Wang and Chu (2020).	This article goes into detail about the performance of Fabric to reveal the varying performances of different ordering services. For this purpose, a network with 20 machines is used, and the different phases of the transaction flow and endorsement policies are considered.
Capocasale et al. (2021).	The authors of this study present a preliminary performance evaluation of Fabric v2.2 and compare it with the performance of Hyperledger Sawtooth. They conclude that Fabric's throughput for non-sequential workloads is considerably better than that of Hyperledger Sawtooth.
Sedlmeir et al. (2021).	This article presents the DLPS benchmarking framework as an alternative to the widely-used Caliper test suite. The authors evaluate the performance of three different Fabric networks and compare it to that of other blockchain implementations.
Thakkar and Natarajan (2021).	This paper examines the performance of Fabric v1.4 with regard to horizontal scaling (e.g., by adding more nodes) and vertical scaling (e.g., by varying the number of CPUs per node). Based on these observations, the authors propose an optimization of the Fabric architecture, including pipelined execution of validation and commit phase.
Toumia et al. (2021).	For this performance evaluation of Fabric, the authors use the Caliper benchmarking framework as well as the fabcar chaincode. The evaluation focuses on the comparison of single ordering service with multi ordering service and considers mixed workloads.
Xu et al. (2021).	The authors of this study developed a theoretical analysis framework to study the performance of Fabric under special consideration of the execute-order-validate logic in Fabric v1.4. By means of a series of experiments, they compare the results with simulations to verify the theoretical model.

Table 1: An annotated bibliography of the literature on performance investigations of Fabric.

concluded that it is not merely the specific infrastructure on which the blockchain resides that has a major impact on performance, but also the design of the transactions and thus their type and number. More recent studies have focused on Fabric v2.0. In particular, Dreyer et al. (2020) have conducted the first measurements of the performance of Fabric v2.0, indicating that it has improved significantly in comparison to older versions of the blockchain framework. More recently still, Toumia et al. (2021) have evaluated the performance of Fabric v2.2. Unfortunately, neither of these studies presents in-depth and comparable results for v1.x and v2.x, since the trial runs were not conducted under the same conditions.

So, although later work introduced further influencing factors, the results of Androulaki et al. (2018) and Thakkar et al. (2018) remain the more complete presentations. Table 2 demonstrates that later work focuses primarily on specific characteristics, such as a sole analysis of the effect of very high network delays. It is important to note, however, that a wide range of other factors has a similarly significant impact on performance, including different benchmarking tools and definitions of key metrics (Sedlmeir et al., 2021), which is why such highly focused studies can only indicate certain trends and first insights into partial developments but are difficult to integrate into the results of other researchers.

In summary, while the literature published to date has provided some significant first insights into the properties of Fabric, these insights have been partial as the literature's parameters are still defined in rather narrow terms. It is also important to realize that the results presented to date are only reproducible to a somewhat limited extent because the methodologies by which they were attained have often been accompanied by minimal descriptions. There is, then, considerable room to improve the general validity of these results.

E.4 Evaluation framework

To advance the current understanding of private permissioned blockchains and conduct further analysis of the multiple variables that may impact the performance of Fabric, we opted for standardized benchmarking. Having examined the many tools used for blockchain benchmarking in the literature we analyzed, we found that those best suited to Fabric are Blockbench (*Blockbench repository* 2022), Caliper (*Hyperledger Caliper Repository* 2022), and the DLPS (Sedlmeir et al., 2021). Blockbench and Caliper, however, do not adequately speficy how they define and determine the key performance metrics, particularly throughput and latency. Since the algorithm by which these are deter-

Source	Fabric version	Vertical scaling	Horizontal scaling	Database	Private data	Multiple workloads	Network delays	Crashing nodes
This paper	2.0 (1.4)	✓	✓	both	✓	✓	✓	✓
Pongnumkul et al. (2017)	0.6	✗	✗	LevelDB	✗	✗	✗	✗
Androulaki et al. (2018)	1.1	✓	✓	LevelDB	✗	✓	✓[2]	✗
Baliga et al. (2018a)	1.0	✗	✗	LevelDB	✗	✓	✗	✗
Dinh et al. (2018)	0.6	✗	✓	LevelDB	✗	✗	✗	✗
Hao et al. (2018)	1.0	✗	✗	n/a	✗	✗	✗	✗
Nasir et al. (2018)	1.0 (0.6)	✗	✓	both	✗	✗	✗	✗
Thakkar et al. (2018)	1.1	✓	✓	both	✗	✓	✗	✗
Koushik et al. (2019)	n/a [3]	✗	✓	n/a	✗	✗	✗	✗
Kuzlu et al. (2019)	1.4	✗	✗	CouchDB	✗	✓	✗	✗
Nguyen et al. (2019)	1.2	✗	✗	n/a	✗	✗	✓[1]	✗
Dabbagh et al. (2020)	1.4 (1.1–1.3)	✗	✗	n/a	✗	✗	✗	✗
Dreyer et al. (2020)	2.0 (0.6/1.0)	✗	✓	n/a	✗	✓	✗	✗
Geneiatakis et al. (2020)	1.1	✗	✓	CouchDB	✗	✓	✓	✗
Wang and Chu (2020)	1.4	✗	✓	n/a	✗	✓	✗	✗
Capocasale et al. (2021)	2.2	✗	✗	n/a	✗	✓	✗	✗
Sedlmeir et al. (2021)	1.4	✗	✓	both	✗	✗	✗	✗
Thakkar and Natarajan (2021)	1.4	✓	✓	n/a	✗	✓	✗	✗
Toumia et al. (2021)	2.2	✗	✓	CouchDB	✗	✓	✗	✗
Xu et al. (2021)	1.4	✗	✓	n/a	✗	✓	✓	✗

[1] The authors only considered delays of 1,000 ms and more, which is far more than the delays that typically occur in a worldwide distributed system.

[2] The authors only considered network delay in a single setting, without stating the actual delay between the data centers involved and without further analysis of the impact of different delays.

[3] The authors did not state the exact version of the investigated Fabric SDK. However, based on the description of the system, we assume this to be Hyperledger Fabric ≥ v1.0.

Table 2: Evaluation of the measurements conducted for the research papers discussed in the above bibliography.

mined remains unclear, we opted for the open-source framework DLPS. This had the added advantage that DLPS allows for sophisticated network deployment with the use of cloud services, which enabled us to test an unprecedented range of configurations.

Our benchmarking covers all the variables identified in the above review process (see Table 2). By conducting tests that went beyond those prior studies, we identified seven additional variables with the potential to impact the performance of Fabric. Since the DLPS did not cover all the particularities of Fabric, our first order of priority was to upgrade the Fabric version supported by DLPS to Fabric 2.0 and include multi-channel setups. Our second extension was to add support for private transactions and complex queries. By way of a third amendment, we extended the supported architectural parameters in such a way as to allow the CouchDB and ordering node docker containers to run either on the same node as the peers or on separate nodes. In doing so, we accounted for the possibility that splitting tasks on multiple machines or joining them to reduce cross-instance latencies might help to increase performance. Our fourth change was to add support not only for simulating network delays but also for multi-datacenter deployments. Our fifth and final improvement was to refine the overall benchmarking process,

Group	Design choices	Answered questions
Architecture (Sec. E.5.2)	Number of organizations, peers, and orderers	How does an organization's configuration of its peers and orderers impact the performance of Fabric?
	Endorsement policy	What is the impact of different endorsement policies?
	Number of channels	Does changing the number of channels impact performance?
	Database location	Does the separation of the database from the peer core functions improve the performance of Fabric?
Setup (Sec. E.5.3)	Hardware	What is the impact of different computer specifications, particularly CPU?
	Database type	How do different databases for the ledger state, such as CouchDB or LevelDB, impact the performance of Fabric?
	Block parameters	How does the choice of blocksize and blocktime affect performance?
Business logic (Sec. E.5.4)	Private data	How does using private transactions impact the performance of Fabric?
	I/O-heavy workload	How do transactions that trigger I/O-heavy chaincode impact the performance of Fabric?
	CPU-heavy workload	How do transactions that trigger CPU-heavy chaincode impact the performance of Fabric?
	Reading vs. writing	What are the essential differences between read and write performance?
Network (Sec. E.5.5)	Delays	To what extent do network delays impact performance?
	Bandwidth	What are the bandwidth requirements for different architectures and throughputs?
Robustness (Sec. E.5.6)	Node crashes	How do crashing nodes impact the performance of Hyperledger Fabric?
	Temporal distribution of requests	How do changes in the temporal request distribution affect the performance of Fabric?

Table 3: Design choices and network specifics in need of thorough analysis.

evaluate single-core CPU usage, analyze traffic stats, and add capabilities to trigger automatic crashes of orderers and peers. To name but one example, this required the dynamic identification of the current leader in the RAFT ordering service. Consequently, the final framework allows testing for all previously mentioned variables found in prior research and extends them with new unique features that, according to our literature review, had not been investigated to date. Table 3 provides a description of all variables considered for this benchmarking. With the publication of this paper, we make our improvements to DLPS, as well as the configurations and results of all the experiments we conducted for this study, available on the DLPS GitHub repository (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021).

We performed the testing in an incremental manner to ensure the reliability of our results. The left chart in Figure 4 describes a single benchmarking run. We used a series of these runs to create a benchmarking ramping series (see the right chart in Figure 4). We also created a configuration file that specifies all particularities of the Fabric network. The DLPS uses this file and automatically sets up a blockchain and client network in Amazon Web Services (AWS) prior to the benchmarking process. A single benchmarking run in the DLPS comprises the sending of requests from clients to the network for a specific duration at a specific rate f_{req}, namely the slope of the requests (see orange requests curve). The receipt of confirmations that the transactions have been processed successfully is illustrated by a response curve (see green response curve). In this curve, one can see how distinguished blocks result in steps as they mark a quasi-simultaneous confirmation of the included transactions. As the graphic shows, the linear response regression describes a curve with a slope that corresponds to the average rate of responses. The average time between sending and receiving confirmation of a specific transaction marks the latency. Likewise, as long as the linear regressions remain parallel, the distance between where the regression for the request and response curves intersect with the x-axis marks the latency.

By starting at a low request rate and repeating tests at an increased request rate in case the network can process requests at the given rate (see x-axis of right Chart in Figure 4), we can localize the maximum throughput, where another increase in the request rate does not improve the response rate any further or indeed deteriorates it due to queueing or overstress. In Figure 4, this behavior can be observed in the right chart at a request rate of approximately 450 tx/s. In each case, we monitored the effectivity – the rate of transactions that were successfully operated – along with controlled resource stats, such as CPU usage and network traffic, to gather any additional information that might help to locate the bottleneck. More information about the DLPS can be found on GitHub (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021) and in the associated paper (Sedlmeir et al., 2021).

By default, the deployments and tests done with the DLPS are highly homogeneous and symmetric. We associate each client with one blockchain node, while each blockchain node is associated with the same number, typically larger than one, of clients. This makes it possible to send requests completely uniformly, which is to say that at f transactions per second, one transaction is sent each $\frac{1}{f}$ second, which is again uniformly distributed among the clients. For example, if we have a 10-node network and 20 clients, and a request rate of 100 tx/s, every client sends requests at 5 tx/s. Moreover, we ensure a uniform offset between the clients. In case a client has multiple cores, and we use multiple workers for multi-threading, we make sure that there is a homogeneous offset, too. It is worth noting that at high request rates, the offset is harder to enforce and far less relevant. Meanwhile, a high degree of uniformity is relevant if one is to measure maximum throughput correctly when it is low, as only then there are no spikes in the nodes' workload.

We used instances from the m5 series in AWS because they strike a good balance between computation, networking, and disk operations, all of which are necessary for blockchain

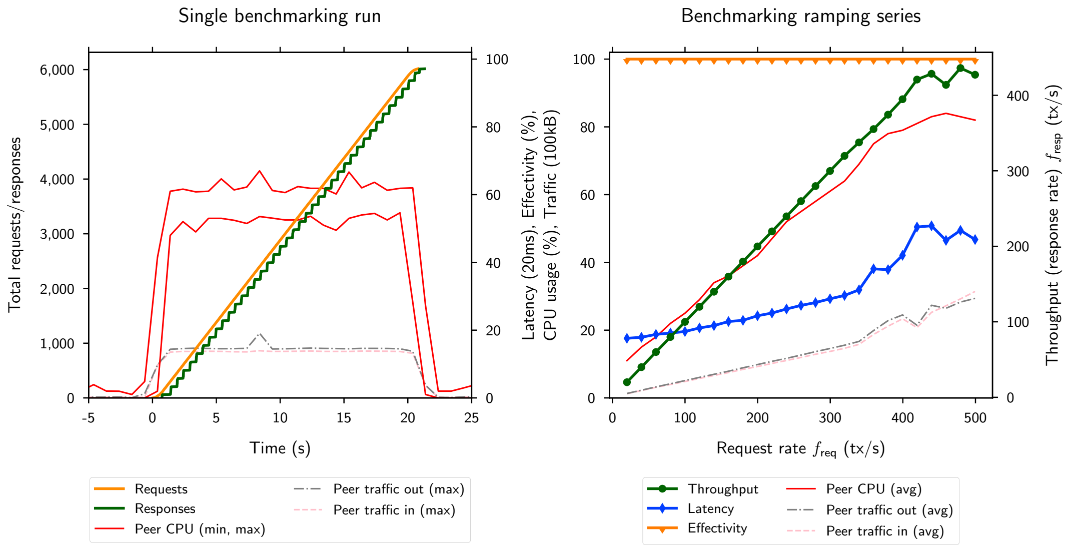


Figure 4: Exemplary benchmark run charts.

nodes. Table 4 features details on these instances. The fully automatic setup with the DLPS takes a minimum of 10 minutes, a reasonable test approximately an hour. We found that reducing the duration of a test increased the variance of its results and tended to overestimate the performance in our tests. Generally speaking, any modifications of network parameters require the blockchain to be completely restarted.

Bearing this in mind, we decided to use a small network (four organizations, each with two peers, one orderer, and four clients) with AWS m5.large instances as our default parameters. We also decided to vary different individual or small subsets of parameters, starting from this default, to keep costs and time within reasonable bounds. Figure 5 illustrates the most significant of the remaining default parameters for the Fabric architecture, while Figure 6 gives an overview of the benchmarking settings. In total, then, our default configuration comprises eight peers and four orderers, as well as 16 clients, in a one-channel network with RAFT consensus. At the start of our experiments, the latest Fabric version was v2.0, which is why we conducted all experiments with this version. When v2.2 was released, we also made spot checks for the purpose of validation and cross-referencing but noticed no significant performance changes. The remaining parameters are described in detail in the dedicated DLPS repository (Fraunhofer FIT and Universities of Bayreuth and Luxembourg, 2021).

In total, our experiments involved near enough 2,000 hours of testing, setting up approximately 1,500 Fabric networks with a total of around 20,000 nodes and 40,000 clients, and sending more than 200 million transactions. As part of this process, we also collected 100 GB of log files to record such factors as the send and response times of each transac-

Name	vCPUs	Memory (GiB)	Network (Gbps)	Storage (Mbps)
m5.large	2	8	Up to 10	Up to 4,750
m5.xlarge	4	16	Up to 10	Up to 4,750
m5.2xlarge	8	32	Up to 10	Up to 4,750
m5.4xlarge	16	64	Up to 10	4,750

Table 4: Used instance types in the AWS m5 series, all based on Intel Xeon® Platinum 8175M processors (up to 3.1 GHz). We added 16 GB of SSD storage. As the operating system, we used Ubuntu 18.04 LTS. Source: AWS (2021).

```
{
  "node_type": "m5.large",
  "fabric_version": "2.0.0",
  "fabric_ca_version": "1.4.4",
  "thirdparty_version": "0.4.18",
  "channel_count": 1,
  "database": "CouchDB/LevelDB",
  "external_database": "False",
  "internal_orderer": "False",
  "org_count": 4,
  "peer_count": 2,
  "orderer_type": "RAFT",
  "orderer_count": 4,
  "batch_timeout": 0.5,
  "max_message_count": 1000,
  "absolute_max_bytes": 10,
  "preferred_max_bytes": 4096,
  "tls_enabled": "True",
  "endorsement": "OutOf(2, 4)",
  "private_fors": 2,
  "log_level": "Warning",
  "client_type": "m5.large",
  "client_count": 4,
}
```

Figure 5: Default settings for the Fabric network architecture.

```
{

  "duration": 20,
  "localization_runs":2,
  "repetition_runs": 0,
  "method": "writeData",
  "mode": "public",
  "shape": "smooth",
  "delay": 0,
  "r2_bound": 0.9,
  "frequency_bound": 100,
  "latency_bound": 10000,
  "delta_send": 0.5,
  "delta_receive": 0.5,
  "success_bound": 0.8,
  "retry_limit": 2,
  "ramp_bound": 2,
  "success_base_rate": 0.8,
  "success_step_rate": 0.04,
  "failure_base_rate": 0.8,
  "failure_step_rate": 0.04,
  "delta_max_time": 10

}
```

Figure 6: Default settings for the benchmarking logic.

tion as well as multiple resource stats such as CPU, memory, disk usage, ping, and traffic for each node and client.

E.5 Benchmarking results

E.5.1 Stability of the default setup and comparison of software versions and databases

Our first comparative analysis of the variously modified default architecture focused on relevant parameters we identified in the above literature review and in our experience of working with the DLPS (see Figure 7). The error bars and areas in the charts in the following figures, which we created with seaborn (Waskom, 2021), represent the standard deviation obtained from conducting every experiment three times, and as these error bars indicate, the results are highly consistent and reproducible. Our default workload is a set of “simple” transactions which consist of writing a single key-value pair to the blockchain’s database, each pair the size of a few bytes.

Confirming the results of Thakkar et al. (2018), we found that write throughput for the default setup with LevelDB is around three times the maximum throughput with CouchDB. Furthermore, we were able to extend this result to private transactions (see Figure 7). We observed that throughput was impacted only at an insignificant rate by the following modifications: doubling the number of clients so as to distribute client workload to more workers (0 % impact on performance with public transactions and 4 % increase of performance with private transactions), doubling the number of channels (2 % decrease of performance with public transactions and 13 % increase of performance with private transactions), deactivating TLS and switching to a centralized (“solo”) orderer (13 % increase on performance with public transactions and 3 % increase of performance with private transactions). This indicates that the bottleneck of the default architecture in our setup is neither the number of clients and channels nor the ordering service and TLS. While the results of Thakkar et al. (2018) lend further credence to our conclusion that the ordering service is not a bottleneck in a similar architecture, they find that doubling the number of channels considerably increases CPU utilization and with it throughput. In our case, however, CPU utilization by peers is already very close to the maximum, and this is true on all virtual cores with one channel. This observation indicates that the two-channel configuration does not, ultimately, exhibit a higher throughput. In our case, the difference between single-channel and dual-channel setup was small, with a variation of only 9 % for private transactions. Public transactions did not even show any significant impact. It is worth noticing, however, that these numbers represent the results with CouchDB. With LevelDB, the relative deviations tended to be even smaller.

Performance benchmarks with older versions of Fabric, particularly those set by Pongnumkul et al. (2017), Dinh et al. (2017), and Nasir et al. (2018), generally yield lower throughput (few hundred tx/s with LevelDB) on considerably better hardware, indicating that the evolution of Fabric has already led to considerable performance improvements. It came as a surprise, therefore, that we noticed a slight decrease in the performance of v2.0, compared to that of the previous version 1.4.4 for CouchDB, as opposed to the results of Dreyer et al. (2020). Indeed, v1.4.4 using CouchDB was about 26 % faster with public transactions and 68 % faster with private transactions than v2.0. With LevelDB, the difference for private transactions dropped to a mere 11 %, and with public transactions v2.0 was 5 % faster than v1.4.4. We put this discrepancy between our results and those of Dreyer et al. (2020) down to how they arrived at their conclusion

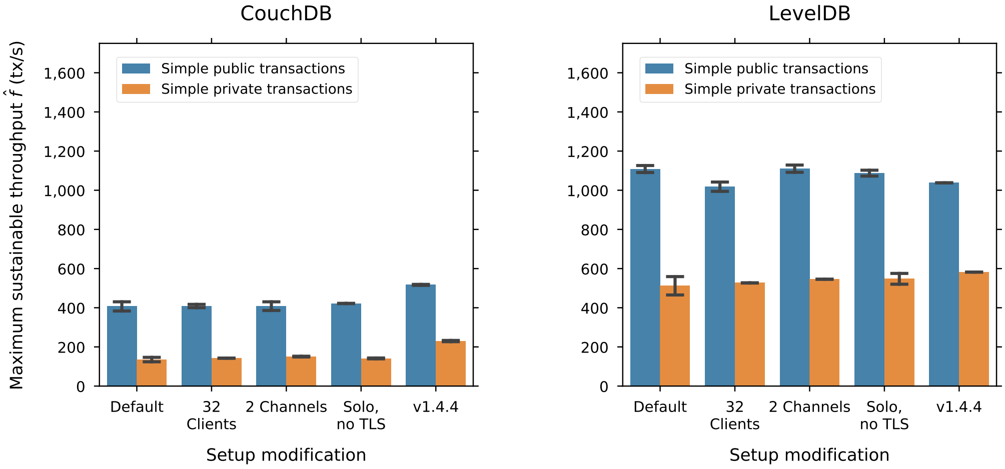


Figure 7: A side-by-side comparison of different architectures in comparison; the configuration for the default setup is described at the end of Section E.4.

since they compared their results for v2.0 with the results of Nasir et al. (2018) for v0.6 and v1.4. However, the testing environment of their studies was different, and Dreyer et al. (2020) used stronger machines with more computing power, which is most likely why they measured a better performance with regard to v2.0. In contrast, our comparison of v1.4.4 and v2.0 was conducted under otherwise identical conditions.

E.5.2 Architecture

E.5.2.1 Endorsement policy

The endorsement policy, described in Section E.2, is a key setting as it drastically changes the level of redundancy in simulation (execution). For instance, with a rise in the number of endorsers, the overhead increases notably, but so too does the robustness. As illustrated in Figure 8, an increase in the number of endorsers leads to the expected corresponding decrease in throughput. In absolute and relative numbers, LevelDB suffers from a much higher performance decrease with a higher number of orderers compared to CouchDB. For example, maximum throughput for simple public transactions with LevelDB decreases by 24 % respectively 54 % when switching from only one endorser and, thus no cross-checks of correct chaincode execution, to two or four endorsements, respectively. For CouchDB, degradation is 14 % respectively 41 %.

For private transactions, we looked at pairwise private collections, which is to say private transactions between two organizations. We found that moving from two to four endorsers results in a loss of 14 % (CouchDB) and 30 % (LevelDB) in maximum throughput. These

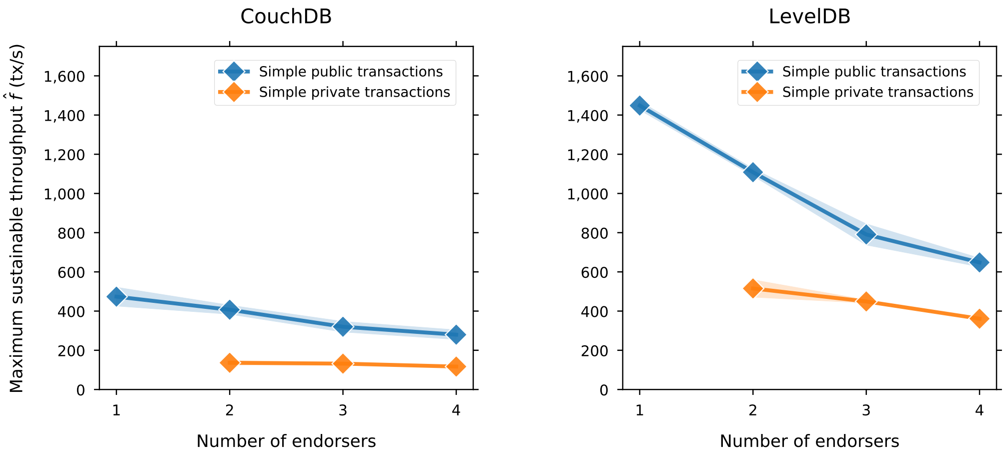


Figure 8: Varying the endorsement policy.

numbers are notably lower than those attained with public transactions (31 % and 42 %). Accordingly, when more endorsements are necessary, performance decreases more for LevelDB, both in relative and absolute terms. Surprisingly, however, one endorser of private transactions proved to be an outlier of sorts. Indeed, we noticed a somewhat strange behavior of Fabric which resulted in throughput in the one-digit range as soon as multiple clients were requesting transactions from different peers. So far, however, we have not been able to determine the reason for this anomaly.

E.5.2.2 Network architecture Initially, we see an increase in maximum throughput when increasing the number of peers per organization while keeping the number of endorsers constant (see Figure 9). Likewise, increasing the number of organizations while keeping the number of peers per organization and the number of endorsers constant increases maximum throughput. However, we also notice that maximum throughput decreases again for large network sizes, so there seems to be an optimum. For the given setup, this optimum is at eight peers per organization and an endorsement policy of two out of eight. We could, therefore, improve public transaction performance by up to 32 % by adjusting the number of peers. With private transactions, the improvements are a little less impressive yet still significant at a 21 % performance increase with eight peers per organization rather than with two peers per organization.

Scaling the number of organizations and the endorsement policy up proportionately only slightly reduces throughput for smaller networks, a potential reason being that the endorsement workload for each peer remains constant and other operations like networking and committing are not the bottleneck in this regime. Nevertheless, for larger network sizes, throughput degrades considerably, and we also see that the difference between hav-

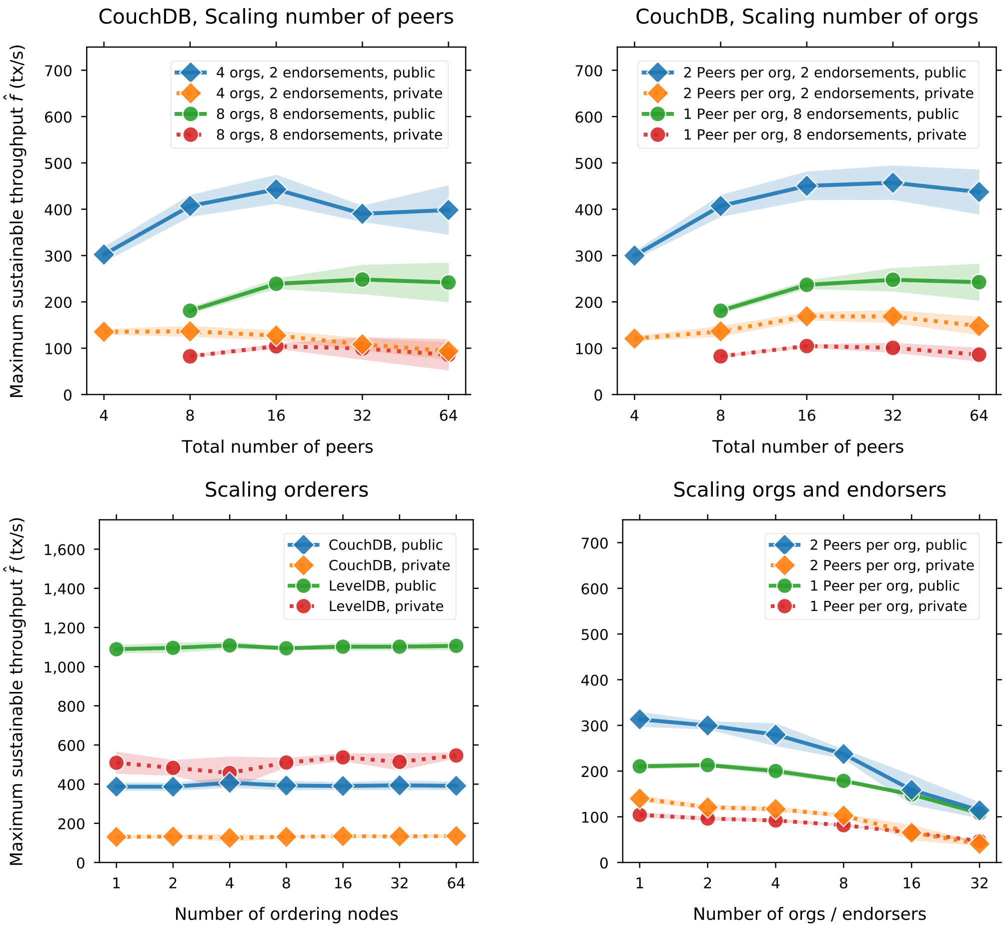


Figure 9: Different scalability parameters in comparison.

ing one and two peers per organization on throughput becomes negligible. This makes sense, as networking becomes the bottleneck in this regime, and adding further peers inside an organization only further reduces the (insignificant) endorsement workload for each peer. Moreover, it increases the already significant networking effort for the anchor peer who receives blocks from the ordering service and distributes them further to the other peers in the associated organization.

For scaling the number of RAFT orderers, we expected a performance decrease but none was yet to be observed in our chosen scenarios. It would appear that, below a request rate of 1,500 tx/s, the ordering service is not a bottleneck for up to 64 orderers, althought it might become a bottleneck for larger ordering services. Using a RAFT ordering service with up to 64 nodes should be sufficient in practically any scenario since this would allow a total of 31 crashes and still ensure the network's functionality.

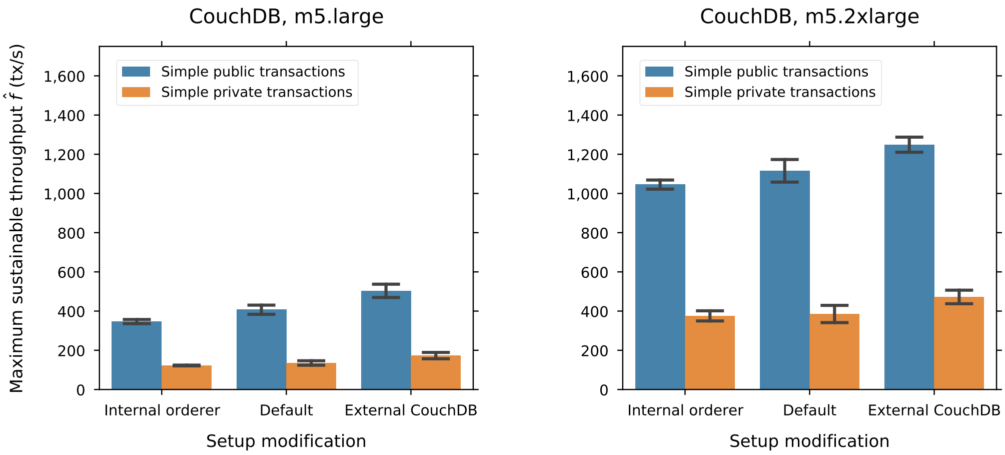


Figure 10: The effect of separating the ordering nodes and the database for CouchDB.

E.5.2.3 Database location Deploying databases, orderers, and peers to separate systems facilitates a small boost in performance (see Figure 10). In our default scenario, the database runs on the peer node (which is obligatory for LevelDB) while orderers work on separate nodes. Our results indicate that running both the orderer and the peer on the same node causes only a slight decrease in performance. Disregarding other important factors, separating an organization's Fabric components on several servers is notably less efficient. In particular, we observed a decrease of only 15 % in the case with the m5.large machines and 6 % in the case of the m5.2xlarge machines. Meanwhile, running the CouchDB on a separate node has a considerable impact on weaker hardware – an increase of 23 % with m5.large. The throughput improvement is similar for private transactions on m5.large and m5.2xlarge hardware, amounting to 22 % respectively 18 %, but it becomes smaller (in relative terms) as soon as better equipment is used (an increase of 12 % with m5.2xlarge).

E.5.3 Setup

E.5.3.1 Database type

Early on during our experiments, we realized that Fabric's performance is contingent on the choice of database. On average, throughput was two to three times higher with LevelDB than with CouchDB. We also conducted individual measurements for LevelDB and CouchDB on the hardware of our default setup and noticed that for writing a single key-value pair, LevelDB has a throughput of more than 5,000 tx/s, while a standalone CouchDB manages only around 500 tx/s. This suggests that in both cases, the databases alone are not the bottleneck, but the individual inefficiencies of those databases have a considerable performance impact. Moreover, CouchDB runs in an individual docker container, whereas LevelDB is integrated into the peer's docker container, which is why the interaction may contribute to the performance differences.

For private data, the difference between the database types tends to be even more noticeable. For example, with private data and m5.large machines, Fabric was 272 % faster with LevelDB than with CouchDB in terms of throughput in the default setup. This confirmed our intuition because a private transaction implies additional write transactions on peers that participate in a private transaction: The payload hash is distributed to all peers in the network, so this involves as many write transaction on each peer as a "normal" transaction would. However, legitimate peers also query the private data from the endorsing peers and add them to their database in an additional write transaction (see also the description of the private transaction data flow in Section E.2.3).

E.5.3.2 Hardware We found it to be important to determine the correlation between machine strength and performance since systems should scale with better hardware (and better network). As long as there are only a few vCPUs, an increase in their number improves performance notably (see Figure 11). For example, when moving from m5.large to m5.xlarge instances, the performance increase for private transactions with CouchDB is 97 % and 62 % when moving from m5.xlarge to m5.2xlarge instances. Similar margins can be observed for both CouchDB and LevelDB and both public and private transactions. However, the improvement made by moving from m5.2xlarge (8 vCPUs) to m5.4xlarge (16 vCPUs) is rather small (less than 25 % for CouchDB and less than 20 % for LevelDB for both public and private transactions), particularly when one takes into account that this also involves twice the costs for hardware and cloud services. When we took measurements with m5.8xlarge instances, we noted that performance improvements were even lower than for moving from m5.2xlarge to m5.4xlarge. Besides, crashes of peers became quite frequent, particularly for LevelDB, which led to our maximum throughput results to be even worse than those for m5.4xlarge instances. Identifying the reasons for this behavior promises to be an interesting starting point for future improvements of Fabric and should allow for better scaling with hardware.

Like Thakkar and Natarajan (2021), we also observed that CPU utilization drops for hardware with a high number of cores. Thakkar and Natarajan (2021) further argue that throughput can be increased by using more peers on multiple channels. However, this is basically the same as running multiple blockchains instead of one, but at present only

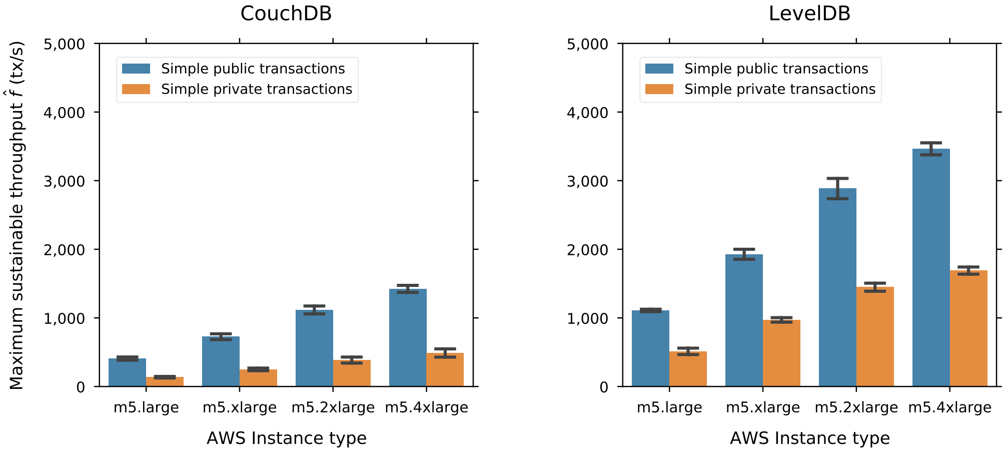


Figure 11: A comparison of different instance types for simple public and private transactions with CouchDB and LevelDB.

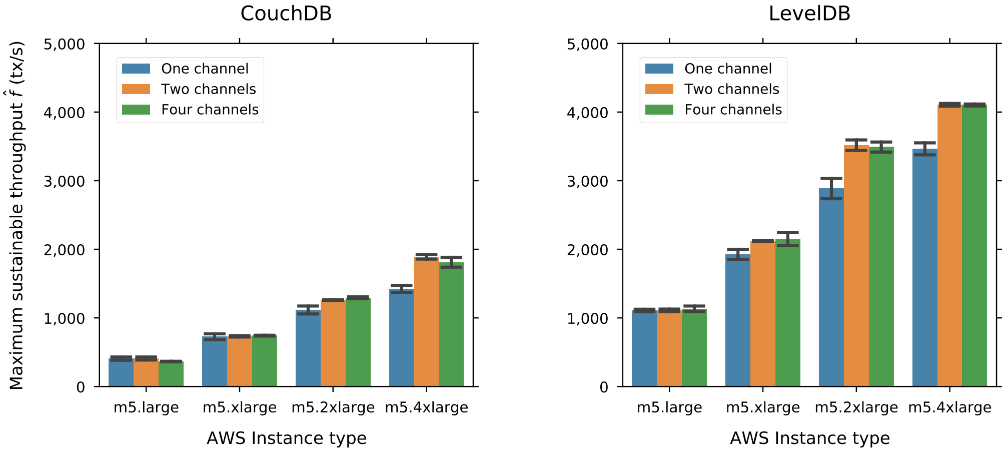


Figure 12: Using multiple channels with varying hardware for simple public transactions with CouchDB and LevelDB.

cross-chain read-operations between the blockchains (channels) are supported. Furthermore, our experiments suggested that for hardware with many cores, the CPUs cannot be fully utilized, and there is not a single core that reaches more than 90 % CPU utilization. Therefore, the computational tasks seem well parallelized and suggest that, ultimately, writing to disk during the commit phase may be the bottleneck. Nevertheless, we were keen to ascertain whether using multiple channels could leverage additional resources. As our results indicate, this does, indeed, seem to be the case, but only to a small extent. Our results (Figure 12) confirm that an increasing the number of channels has a small impact (an average of 12 %, regardless of the database type) when switching from a single-channel setup to a dual-channel setup. Adding further channels leads to no noticeable further improvement in maximum throughput.

E.5.3.3 Block parameters New blocks are generated by the ordering service when either the maximum blocksize is reached or the time that has passed since the generation of the last block is longer than the blocktime. For the purpose of our experimental setup, we selected a request rate of 500 tx/s, at which we observed that the response rate (throughput) cannot exceed 500 tx/s, yet it will be around 500 tx/s when the blockchain can handle at least 500 tx/s. Since the maximum blocksize is 1,000 transactions, and the blocktime is two seconds, this means that blocks cannot comprise more than 1,000 transactions. Therefore, it will always be blocktime that triggers the creation of a new block. By the same logic, too great a reduction in blocktime results in a throughput decay caused by the block production overhead. Also Thakkar et al. (2018) have noted this positive correlation between block size and maximum throughput. For larger blocktimes, the transaction workload dominates, which is why performance tends to be far less contingent on changes in blocktime. Latency naturally increases with blocktime, as it is always the associated timeout that triggers the creation of new blocks. On the other hand, decreasing maximum blocktime by decreasing latency below 0.5 s also heavily decreases throughput. Consequently, we find that a block timeout of around 0.5 s constitutes a sweet spot – any decrease would make throughput significantly worse, and any increase does not substantially improve throughput yet increases latency.

When we varied maximum blocksize, we got similar results, but with a “cutoff”. This is because we used the default maximum blocktime of 0.5 s, which – considering that maximum throughput is around 500 tx/s when blocks become sufficiently large – becomes the actual trigger as soon as the maximum blocksize is higher than $0.5\,\text{s} \cdot 500\,\text{tx/s} = 250\,\text{tx}$. For the low throughput tests on latency, i.e., at 50 tx/s for public transactions and a blocktime of 500 ms, blocks never get bigger than 25 tx. Accordingly, we see no changes in latency beyond 50 ms. See Figure 13 for an overview of these results.

E.5.4 Business logic

E.5.4.1 I/O-heavy workload

The first test we ran was on the impact of maintaining larger data sets in terms of the keyspace size of the state databases. We did not observe any relevant dependence on the keyspace size for less than 10^5 keys (see Figure 14). Performance implications of very large keyspace sizes for LevelDB are given by the likes of Baliga et al. (2018a) and Dinh

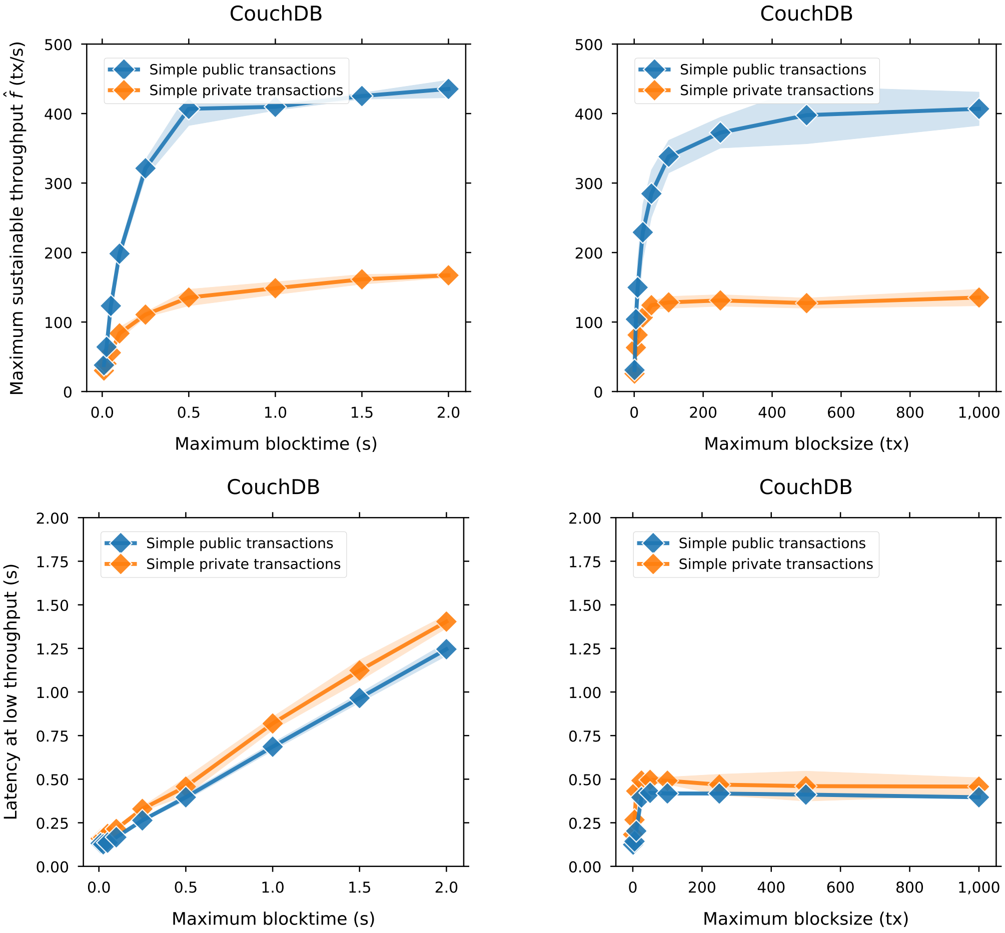


Figure 13: Comparison of different block times and block sizes.

et al. (2017). Due to space restrictions, we consider this to be a property of the databases themselves rather than the Fabric network.

Next, we checked how sensitive throughput reacts to changes in size of the data written in a single transaction, both when the data is communicated via the client (data sent from the peer) and when it is already present on the peer (for instance, created there a result of executing a smart contract). We observed that, as long as the bandwidth is adequate, like in a cloud data-center, it is not significant whether a large amount of the data that is to be processed is created on the peer or sent via the client. Transactions with 10 bytes have around the same throughput as the simple (public/private) transactions that have already been benchmarked before. Switching from 10 bytes to 1 kB only causes degradations of less than 10 % for CouchDB and less than 20 % for LevelDB. However, moving from 10 bytes to 100 kB degrades throughput by more than 85 % for public transactions (even 95 % if the data is generated on the client; which indicates that networking is also resource-intensive) and 75 % to 95 % for private transactions. We also noted that

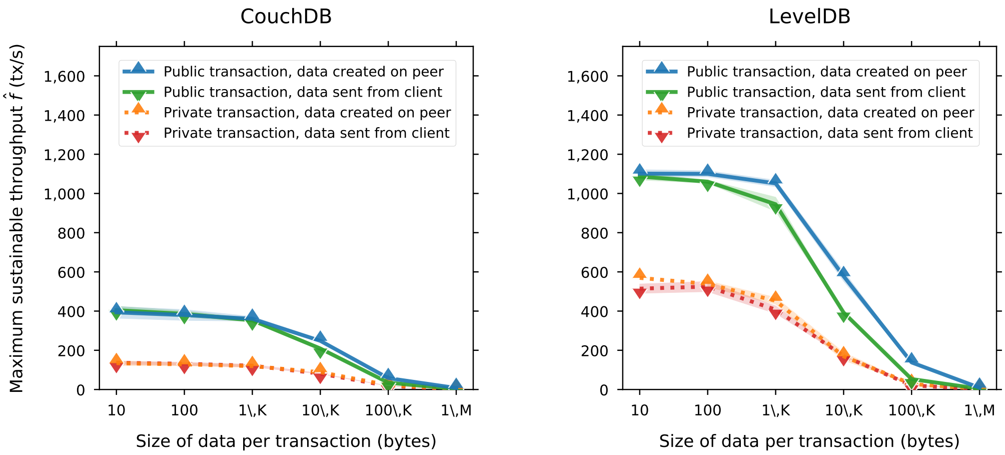


Figure 14: Comparison of performance with varying transaction size.

the degradation of CouchDB and LevelDB is similar, except for private transactions with CouchDB. Here, throughput is already rather low for 10 kB. So, while there is no significant difference between the creation of the data on the client (networking intensive) and peer (no additional networking) for 10 bytes, it is 200 % higher for 100 kB LevelDB public, private and CouchDB public. For private transactions with CouchDB, the difference is only 30 % – 50 %. For 1 MB, throughput is less than 10 tx/s for LevelDB and less than 3 tx/s for CouchDB. The maximum throughput in terms of data is approximately 14 MB/s for the run with 100 kB packages. This low sensitivity of throughput to transaction size up to the order of few kilobytes may be due to the certificate handling of Fabric: Each transaction carries the digital certificates of the client that submits it and the peers that endorse it. We measured that these certificates – including the sub-certificate of the corresponding organization and the root certificate of the certificate authority – have a size of approximately 1 kB. Accordingly, as long as the payload is smaller than 1 kB, it has a negligible impact on peers' and orderers' networking effort.

E.5.4.2 Reading data

First, we checked that the keyspace size does not impact fewer than 10^5 keys. Reading speed is only a reasonable number on a "per peer" basis because no other node is involved in a reading operation (except for cross-checks if the client does not trust the peer). For simple key-based queries on m5.large instances, we obtained approximately 400 reads per second on CouchDB (150 reads per second with complex queries) and around 750 reads per second on LevelDB. We used non-invoked queries that do not involve the Fabric transaction flow from Section E.2. Again, we used the standard configuration, consisting of four clients and two peers, in accordance with which clients distribute requests equally between the peers.

Complex queries are only feasible on CouchDB. Here, we could observe a massive difference between no indexing (which performs approximately as badly as querying the total database and searching the value space afterward, resulting in a low one-digit number of successful queries per peer and second) and indexing, which still allows approximately 150 reads per second and peer. It is worth noting that networks with high-performance requirements on reading processes should either opt for multiple peers for scaling benefits or consider retrieving the peers' data and maintaining a separate database for queries.

E.5.4.3 CPU-heavy workload

To test Fabric's performance on CPU heavy operations, we conducted matrix multiplications which we implemented through simple nested loops, with different matrix sizes as this allows for quantitative control of complexity. Please see Figure 15 for an overview of our findings. Multiplying two n×n matrices requires $\mathcal{O}(\mathrm{n}^3)$ simple operations (additions and multiplications) in our nested loop implementation. So, for large n, we expected the throughput to scale as $\frac{1}{n^3}$. Indeed, we see that the total number of operations approached a saturation curve for large n. In contrast, we found that, for small n, the Fabric-related overhead matters. For n=300, the performance of the network is still around 30 tx/s respectively 15 tx/s for two respectively four endorsements. As we could measure, this corresponds to the performance of a standalone node.js application that runs the nested loops. To be more precise, in Fabric, the chaincode is run in separate docker containers that communicate with the peer container, so every endorsing peer's associated chaincode container executes the code. During validation, the peers then check that the simulation results of the endorsing peers coincide. This means that, leaving aside the degree of redundancy determined by the endorsement policy, the chaincode (smart contract) can run code as efficiently as a centralized system when the blockchain-related operations (networking, signatures) are negligible in comparison to the overhead caused by networking and committing operations in Fabric. In contrast, a matrix multiplication that is also implemented through nested loops and run in the Ethereum Virtual Machine cannot deal with a multiplication of a 90×90 matrix. Furthermore, multiplying a 30×30 matrix takes almost an entire second. With regard to executing CPU-intensive tasks, then, all of these data points illustrate the significant performance benefits of using Fabric compared to (both private and public) Ethereum-based blockchains when executing CPU-intensive tasks.

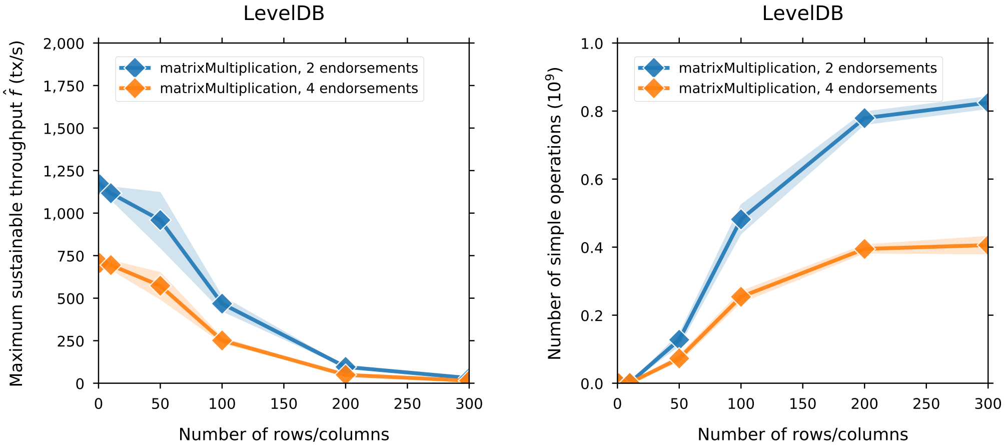


Figure 15: Different tasks difficulties by matrix multiplication.

As expected, we were able to confirm that there is no difference between public and private transactions for a matrix multiplication since there are no database operations involved. Moreover, for a stricter endorsement policy (four out of eight), throughput is approximately half of that attained with a weaker endorsement policy (two out of eight) because, in total, there are twice as many computations for a single transaction. When comparing four endorsements and two endorsements, the ratio of maximum throughput is 40 % for multiplying a 1×1 matrix, 46 % for a 100×100 matrix, and 50 % for a 300×300 matrix. With the initial presence of Fabric-related overhead for small matrices, we hence get the expected asymptotic value.

E.5.5 Network

E.5.5.1 Delays

To investigate the impact of network delays in a sufficiently general real-world scenario, we defined groups within our default architecture, where each group corresponds to an organization, representing an enterprise and consisting of two peers, one orderer, and four clients. We proceeded on the assumption of minimal network delays within a group. While this hypothesis is certainly optimistic for global enterprises, if speed is of the essence in a large network, one may well choose the nearest peers within an organization for endorsement. In a first attempt, we used the standard traffic-control (tc) tool available on Ubuntu servers to set an artificial delay for any communication between the members of different groups. However, we noticed that the results obtained by imposing artificial delays became very unreliable at high throughput, which indicates that, when CPU utilization or network traffic is high, tc does not operate reliably. To address this,

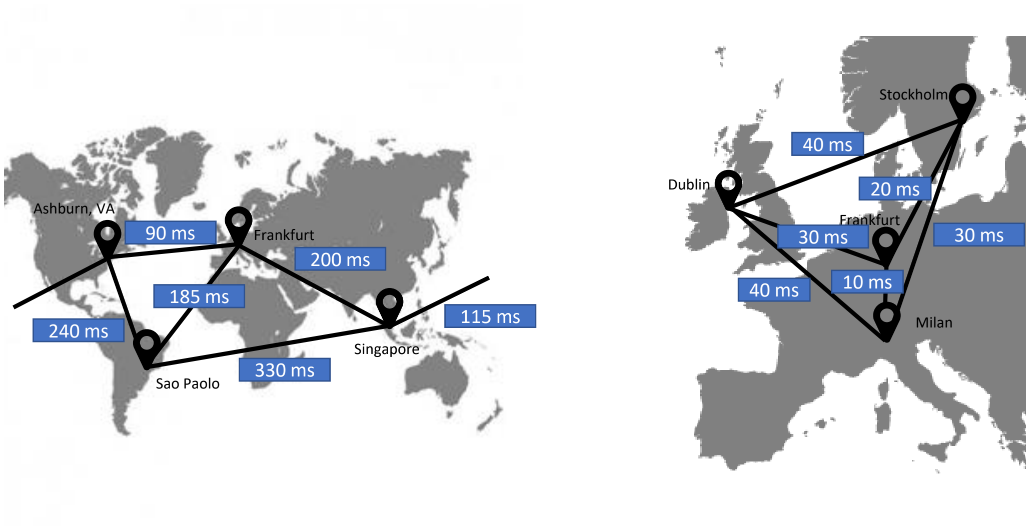


Figure 16: Network topologies and corresponding network delays (one-way) used to determine the impact of network topology on maximum throughput.

we started using deployments over multiple data-centers and set up a cross-European and global network. Specifically, we set up groups in Germany, Ireland, Italy, and Sweden for the European case with moderate network delays, and in Germany, Brazil, Singapore, and the East of the US for the intercontinental case with high network delays. We noted that latency increases by 30 % to 50 % from a single datacenter to a cross-European (30 ms one-way network delay) and by more than 200 % from a single data center to an intercontinental distributed system (up to 330 ms one-way network delay). In the intercontinental case, transactions will, on average, take 1.2 seconds (public) and more than 1.7 seconds (private), even at low throughput. Once throughput approaches maximum sustainable throughput, the latencies become even higher. A detailed topology of the network, including the network delays we measured between each data center pair, can be found in Figure 16.

Having imposed artificial network delays by using tc for our initial simulation, we noted a decrease in performance by approximately 50 % for CouchDB and 70 % for LevelDB as well as a significant standard deviation thereof for delays of 50 ms. Meanwhile, by using the actual cross-data center deployments with real-world delays, we found that, for both LevelDB and CouchDB, and indeed for both public and private transactions, performance does not degrade as significantly in the intercontinental case (see Figure 17). This lends further credence to a statement by Androulaki et al. (2018), according to which a cross-data center deployment of a large number of nodes still offers high performance. In their experiments, the authors deployed 100 nodes, located in five different datacenters, and used LevelDB for the peers' databases. In our own experiments, we found that, for public

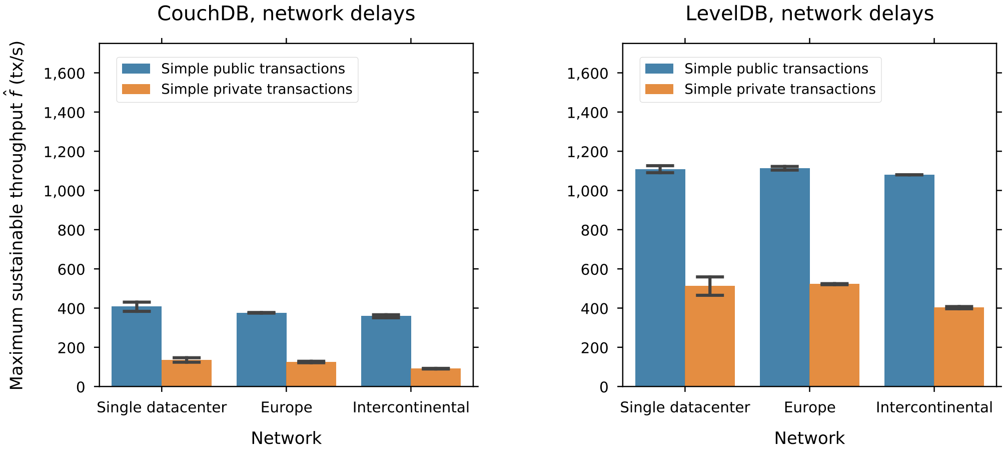


Figure 17: Maximum throughput for different geographical distributions of nodes.

transactions with CouchDB, maximum throughput decreases from 426 tx/s for the single datacenter case to 376 tx/s in the cross-European case and 358 tx/s in the intercontinental case. This corresponds to a drop of 12 % and 16 %, respectively. We also observed that the performance decrease is less significant for LevelDB in the cross-European case. In contrast, for both CouchDB and LevelDB, the throughput decrease of private transactions in the intercontinental case is considerable. Indeed, for CouchDB, we observed a drop of 39 % in maximum throughput compared to the single datacenter case. For LevelDB, we noted a drop of 26 %. We attribute this increased latency sensitivity of private transactions to the additional networking required in order to distribute the payload to the eligible peers (see Section E.2.3).

To conduct a systematic investigation of the relationship between performance metrics and network delays, we had to adapt our benchmarking procedure. The same was required for a latency analysis, which we found to be (intuitively and empirically) far more sensitive concerning network delays than throughput. While the real-world deployments make it difficult to vary network delays continuously, we found that the latencies in those real-world deployments are similar to the latencies we observed when we stayed well below maximum throughput in our measurements. By conducting corresponding experiments with artificial network delays imposed by means of the tc tool, we found that transaction latency increases proportionally with network delays;. Interestingly, the average slope in this case is approximately 15. In other words, an increase in latency Δl for bilateral communication between servers (including clients, peers, and orderers) that belong to different organizations implies an increase of around $15 \times \Delta l$ for the latency of transaction confirmation, i.e., from triggering the transaction request on the client until receiving a confirmation on the client that the transaction has been committed in a

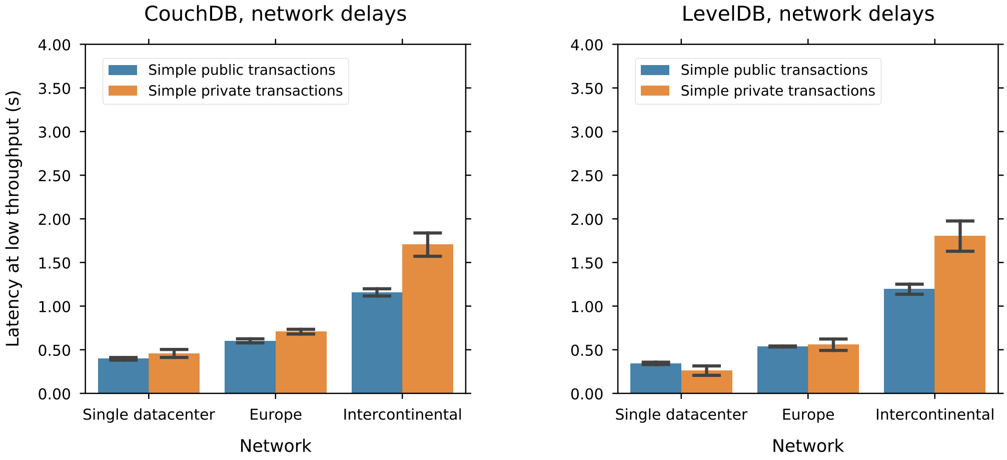


Figure 18: Latency for different geographical distributions of nodes.

peer's ledger. This suggests that there are approximately 15 communications between different nodes in the lifecycle of a single transaction. Since Fabric networks have to meet high performance requirements, it is especially important to avoid communication paths with notable network delays. This can be achieved, for instance, by weakening endorsement policies and choosing endorsers with low network latency, or by avoiding particularly large distances between ordering nodes. Opting for close proximity between nodes, however, can negatively affect their availability guarantees ("liveness") because stronger geographic localization increases the threat of correlated crash-failures as, for instance, caused by blackouts in power grids.

E.5.5.2 Bandwidth

Within a Fabric network, each role requires certain bandwidth, so we investigated those requirements with regard to orderers, peers, and clients. For peers and clients, we found that inbound traffic is distributed uniformly. Moreover, the maximum requirement on download speed is homogeneous for peers and clients, as it is for orderers. The maximum values we observed were the same as the respective maximum on outbound traffic, and since upload speed is more likely to create a bottleneck than download speed, we will focus our discussion on the requirements concerning upload. According to intuition, as Figure 19 illustrates with regard to these three roles in the network and different architectures, there is a general linear correlation between throughput and outbound traffic for all roles. In their Fabric network, Thakkar et al. (2018) measured the download rate of a peer to be approximatel 2.5 MB/s (and the download rate 0.5 MB/s). Regarding the upload rate of peers, we arrive at a similar order of magnitude for equally high throughput.

In contrast, we found the upload rate of orderers to be more heterogeneous, and at times significantly more sizeable. To be specific, the RAFT leader requires a very high upload speed when the ordering service has multiple nodes. For n=64 orderers, for example, we observed an upload rate of more than 350 MB/s (bearing in mind that maximum throughput is independent of the number of orderers for up to 64 orderers, upload is still not the bottleneck, at least not for deployment in a single datacenter with high networking capabilities). This is plausible because the crash-fault tolerant consensus mechanism RAFT, which Fabric uses for the ordering service, follows a two-phase commit paradigm. Therefore, the complexity of network traffic, i.e., the number of sent messages, is in the order of $n \times (n-1)$, and the leader needs to be involved in all of these messages. For the other orderers, outbound traffic is one order of magnitude smaller. As the charts in the second row of Figure 19 indicate, the upload speed of non-leading ordering nodes depends mainly on the number of peers in that network, as well as on the number of endorsers per transaction, which both makes sense as they need to distribute new blocks to the peers. After all, transactions are larger when more endorsements (signatures) have to be collected.

As rows three and four row of Figure 19 indicate, this observation also holds true for the upload requirements of peers and clients. Moreover, for the clients, the linear interrelation between outbound traffic and maximum throughput is clear. The upload speed requirements on non-leading orderers are often approximately twice the requirement on the peers. This makes sense as there were twice as many peers as orderers in our default scenario. It is further worth noting that the clients only have a very small requirement concerning outbound network speed.

E.5.6 Robustness

E.5.6.1 Temporal distribution of requests

As illustrated in Section E.4, the DLPS sends transaction requests highly uniformly by default. When we tested different temporal distributions of the requests (i.e., jitter), we modified this default to a step-shaped distribution in order to evaluate the queuing system's sensitivity and efficiency. Here, clients send transactions at the beginning of each second and they do so with a fluctuating distribution that has notably more or fewer transactions per second ($\Delta \leq \frac{f}{2}$). In this scenario, we did not notice a significant deterioration

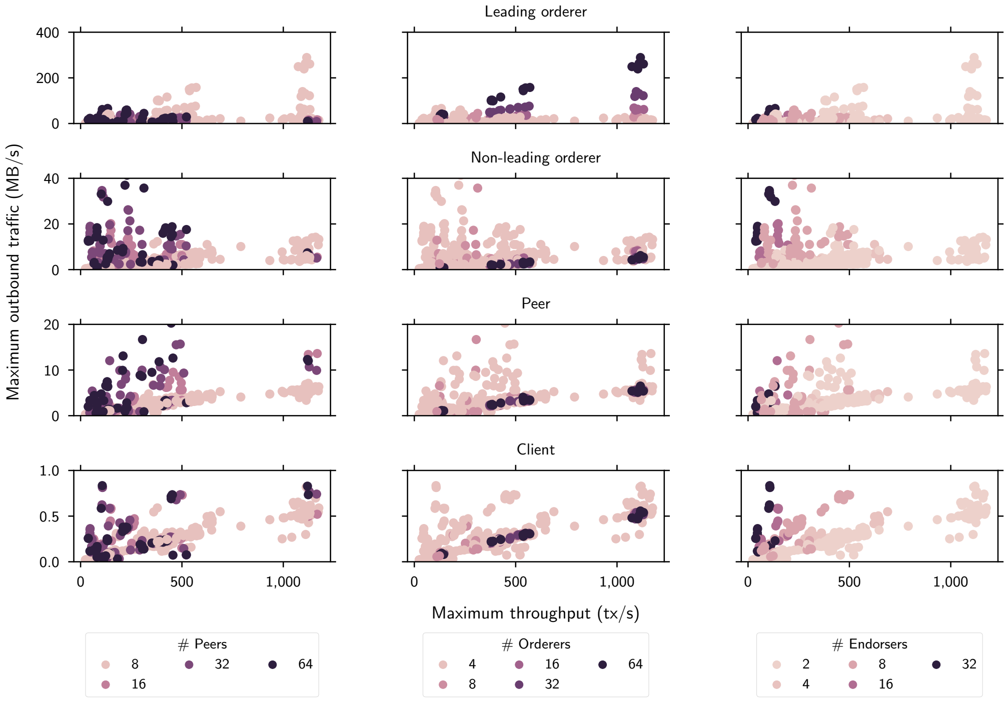


Figure 19: Required bandwidth for different roles and architectures.

of maximum throughput and latency. This suggests that, as long as queues do not become too long, the queuing process of Fabric is efficient.

E.5.6.2 Node crashes

As soon as a system transitions from testing to productive usage, its resistance and resilience against failures become matters of great relevance. By operating multiple peers within one organization on physically separated nodes with the use of a blockchain, the risk of data loss caused by crashes and attacks is already notably mitigated. Similarly important, however, is how the failures of individual nodes affect overall performance, since it might take some time until a failed node is compensated or reset and re-synchronized. For the purpose of this study, we examined the different roles in the system with the expectation that each would have different failure ramifications. We focused on crashes because malicious attacks require sophisticated and specialized implementations and – since we are in a private permissioned network – can be traced to the responsible parties and therefore deterred. While this will no doubt make an interesting topic for future research, we will not look into it any further here, nor into crashes of clients since we used enough clients to saturate the system and clients can easily be replaced on short notice as there is no need for synchronization. Therefore, the relevant aspects for this study are the crashes of orderers and peers. Since the recommended ordering service is the crash

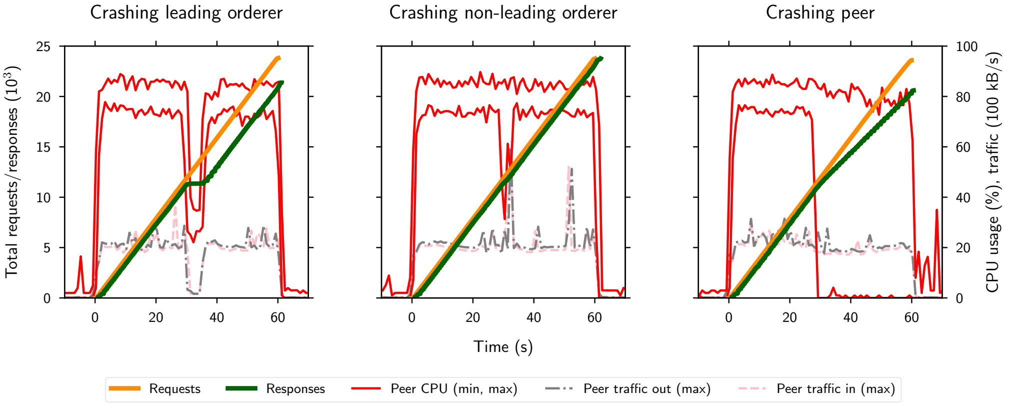


Figure 20: Performance impact of crashing leading or non-leading orderer nodes and peers (at t=30 s) on performance.

fault-tolerant RAFT, we expected that crashing a single orderer would not significantly impact a Fabric network's operation. Figure 20 depicts the impact of crashing various node types.

To test the outcomes, we put the network under stress at 400 tx/s, which is close to maximum throughput (and therefore maximum CPU utilization). After 30 seconds of sending transactions, we crashed a single orderer and continued sending requests at the same rate for an additional 30 seconds. We found that the total impact of crashing an orderer is indeed limited. However, it makes a considerable difference whether the crash affects the current RAFT leader or a non-leading orderer: When the crash affects a non-leading ordering node, the ordering service stops distributing new blocks for around 5 seconds, whereupon it resumes at the previous speed with a newly selected orderer (Figure 20, chart on the left). If a non-leading orderer crashes, however, the impact on performance is negligible (Figure 20, chart in the center). If a single peer crashes, the performance drops by the rate of transactions that required the respective peer as an endorser. However, this is predicated on the fact that we limited clients to requesting endorsements from a fixed set of peers that contained exactly as many peers as required by the endorsement policy. In this case, we used our default configuration with four organizations, each associated with two peers, and two endorsements for every transaction. Consequently, every peer participates in $\frac{1}{4}$ of all transactions, which explains the drop of throughput by 25 % after 30 seconds. In a production-grade Fabric network, one would likely provide each client with at least a few more peers to compensate for crashes, which would prevent failing transactions. However, shifting the endorsement workload to another peer may decrease maximum throughput to that of a Fabric network without the crashed peer. For a comprehensive discussion of other reasons for error-prone transactions, see Chacko et al. (2021).

E.6 Discussion

Fabric is a highly customizable permissioned blockchain framework that allows enterprises to adjust the network architecture to their requirements. While this ability allows for many optimizations, it also leads to complexity and requires in-depth knowledge of Fabric's design options and parameters as well as their performance impact. The purpose of this paper is to link in with current research in efforts to advance the understanding of the key metrics one has to consider when setting up a Fabric-based application. Having managed to reproduce many of the results of these related works, we broadened our expertise in Fabric with a bilateral strategy; by using a benchmarking framework built on precise definitions of key metrics and by testing many as yet unexplored settings in a structured manner. For instance, we worked with the findings of Androulaki et al. (2018) concerning the effect of network delays on the throughput of Fabric, but we extended their results by comparing three different setups: no delay, continental, and inter-continental. This allowed us to understand how DLPS can be used to run tests for a wide range of variables in order to evaluate a blockchain-based system's performance potential prior to implementation. Please see Table 5 for details on each contributing factor.

As the overview of our measurement results in Figure 21 shows, maximum throughput depends largely on the type of transaction (reading operations, CPU heavy transactions, i/o heavy transactions, and simple write transactions) and the type of hardware. For homogeneous hardware (such as m5.large, for which we conducted the most experiments), there is a clear correlation between maximum throughput and CPU use across highly heterogeneous deployments. We found that both are heavily dependent on the kind of database used (LevelDB achieves higher throughput), the visibility of transactions (private transactions achieve lower throughput), and network size (large Fabric networks have lower throughput). Therefore, these parameters should be considered with particular attention to detail when conceptualizing the network architecture for a use case with high performance requirements.

Kannengießer et al. (2020) describe various trade-offs that developers have to make when working on blockchain systems. This study examines some of these trade-offs and provides additional metrics to quantify them in the case of Fabric. In particular, our measurements of different Smart Contract methods, such as varying the complexity of matrix multiplications and the size of transactions, quantify the trade-off between transaction validation speed and operation complexity. Similarly, by investigating private and public transactions in Fabric, we also quantify the trade-off between confidentiality and per-

Group	**Impacting Factor**	**Results**
Architecture	Number of organizations, peers, and orderers	In our experiments, the number of orderers does not affect overall performance. Adding peers to small networks while keeping the endorsement policy constant improves performance. When a large number of organizations gets involved, performance degrades, albeit to a rather small extent at first but increasingly so with bigger networks.
	Endorsement policy	A higher number of required endorsers reduces total throughput. Introducing additional peers can mitigate this.
	Number of channels	The number of channels has little impact on the performance of the system.
	Database location	Moving the database to another server offers limited benefits for throughput.
Setup	Hardware	Increasing the number of vCPUs increases throughput significantly for fewer than eight vCPUs but less for a larger number of vCPUs.
	Database type	The database type has a significant impact on system performance. With LevelDB, throughput is up to three times higher than with CouchDB.
	Block parameters	Block time of around 0.5 s yields a particular sweet spot. Any addition of block time or block size, respectively, has only limited performance benefits but increases block latency. Below a block time of 0.5 s, throughput decreases considerably.
Business logic	Private data	Throughput for public transactions is around three times higher for CouchDB and around two times higher for LevelDB than for private transactions.
	I/O-heavy workload	Once the transaction payload is larger than 1 kB, throughput decreases rapidly. Total upload is bounded to tens of MB/s.
	CPU-heavy workload	CPU-heavy node.js smart contracts work as fast as native implementations.
	Reading vs. writing	Read throughput scales linearly with the number of peers while write throughput depends on many parameters.
Network	Delays	The impact of network delays on throughput and latency is relatively low, even in an intercontinental network. This impact is greater on private data than it is on public data.
	Bandwidth	The bandwidth requirements rise proportionally to the number of nodes. Considering the RAFT setup, the leader node demands a comparatively higher upload bandwidth with an increasing number of orderers.
Robustness	Node crashes	Fabric is very robust with regard to crashes. A crashing peer does not affect the total network beyond its loss in endorsement power. If a Raft leader crashes, it takes about 5 s for the system to resume normal operations.
	Temporal distribution of requests	Small deviations in distribution do not impact system performance as long as peaks stay below the maximum sustainable throughput.

Table 5: Results of our benchmarking study by impacting factor.

formance suggested in article of Kannengießer et al. (2020). Finally, our various performance measurements of varying network sizes, topologies, and endorsement policies

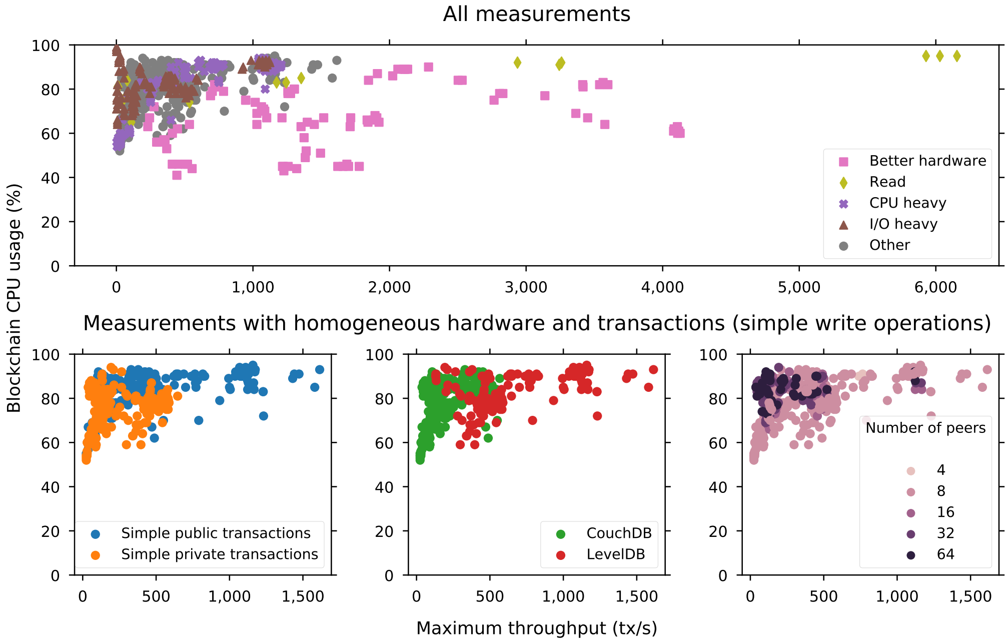


Figure 21: Summary of our measurements and the most important design parameters.

quantify the extent to which performance depends on decentralization and with it security and availability. As our experiments demonstrate, the ordering service is not the bottleneck in the examined architecture, so the trade-off between performance and security was only present in the endorsement policy. The solo orderer, lacking any crash or Byzantine fault tolerance, provided approximately the same overall performance as the crash fault-tolerant RAFT ordering service. It will be interesting to see whether the future use of a Byzantine fault-tolerant ordering service will have any impact.

Our results have several implications concerning the use of blockchain for supply chain management. First, we were able to validate the theoretical performance of Fabric in that it supports up to several hundred or indeed a few thousand transactions per second. Nevertheless, due to the current implementation of gossip dissemination, we found that the performance decreases when Fabric serves a large number of organizations. Private blockchain systems based on Fabric should, therefore, avoid the integration of too many organizations. Second, while private data is an important function of supply chain information sharing systems (Guggenberger et al., 2020), we found that it also reduces overall performance to a surprisingly large extent. With this in mind, we propose to use this function only where necessary. At all other times, standard public data transactions are to be given preference, or Fabric-based applications ought to be supplemented with bilateral communication via standardized APIs to avoid overloading the blockchain. Third, the same applies to the database type. Advanced blockchain provenance solutions can use

CouchDB's query capabilities to quickly extract data from the system. While CouchDB offers fast queries, its use considerably degrades system performance. Therefore, its use is only advisable when absolutely necessary.

Another solution, and one that a future iteration of Fabric could turn into a desirable feature, is that an organization can also keep the world state in another database by periodically pulling an update from the database natively supported by Fabric peers. This other database could be optimized for the queries required by the use case, for example, a graph database for supply chain use cases that record the successive joining of components. This approach would also make it possible to decouple query operations from the peer tasks. Doing so would ensure that no queries could impact system performance. Since large network sizes degrade throughput, another useful approach would be a horizontal scaling of databases associated with one peer, instead of deploying several peers, but this would only be advisable if an application is expected to have a substantial query throughput.

Finally, despite the limitations that became apparent in our benchmarking results, Fabric performs exceedingly well on tasks that require extensive computation in an intercontinental environment. The impact of network delays is limited, which puts a global infrastructure, as proposed by TradeLens (Jensen et al., 2019), within the realm of feasibility.

Here, we have focused on a subset of interesting factor combinations because the large number of parameters that can be configured in a Fabric network makes it impractical to test all possibilities. We settled on a standard configuration and changed individual parameters to identify their impact on performance. It is important to note, therefore, that fellow researchers who depart significantly from our testing scenario might attain somewhat different results. To name but one example of this potential deviation, Thakkar and Natarajan (2021) suggest that certain characteristics might change for very strong servers.

Accordingly, the results of this study are to be understood as indications of the potential of Fabric, rather than as strict reference points for all possible cases. We suggest that those who wish to move ahead to operational systems should first conduct a specific evaluation in which to include their respective particularities. When doing so, companies are welcome to use our extended version of the DLPS framework to test a wide range of variables and verify their blockchain project's feasibility. Ultimately, the suitability of Fabric for business processes is case-specific. For example, we see an opportunity to use Fabric for the tracking of high-value goods, such as medical devices, in a B2B environment. However, other use cases may have requirements that go beyond Fabric's capabilities. In particular, use cases with extreme spikes in transactions may be unsuitable for Fabric. On Singles Day in 2020, for instance, there were periods in which the online Marketplace Alibaba received almost 600,000 orders per second (Forbes, 2020). In its current iteration, Fabric would not be able to process that many orders.

E.7 Conclusion

The purpose of this paper has been to examine the performance of Fabric. It provides an in-depth analysis of the system, covering a total of 15 variables concerning architecture, setup configuration, business logic, network, and robustness. The benefit of this analysis is the guidance it offers system architects and infrastructure engineers when designing Fabric-based infrastructure and applications.

To differentiate between the various theoretical and practical contributions of this study, it is important to note that, from an academic perspective, it contributes to the current understanding of how to design blockchain systems. It does this by exploring the full potential of private permissioned blockchains. We expect that future research will also benefit from the extended list of contributing factors by using it as a baseline for performance analyses of Fabric and other blockchains. As this study has indicated, in-depth performance analyses should incorporate corresponding measurements of the impact of multiple parameters. From a practitioner's point of view, this study should be of value for demonstrating the impact of various parameters. This should help to optimize existing applications so as to facilitate higher network performance. Finally, by discussing the potential of Fabric, we provide a practice-oriented foundation on which practitioners should be better able to understand whether blockchain might meet their requirements for any potential applications at the level of their operations. They may also benefit from using our extension of the DLPS to benchmark their respective chaincodes and determine the feasibility of productive use in networks with specific parameters.

As our results demonstrate, Fabric is suitable to support the needs of many real-world industrial blockchain applications as it provides sufficient scalability along with multiple other key properties, such as ensuring high availability guarantees, accountability, as well as to some extent robustness to manipulation attacks. Owing to the large number of system parameters in Fabric that – according to the measurements that we present in Section E.5 – it is challenging to provide a universal and succinct quantitative assessment of Fabric's performance; and a pre-study that considers the specific parameters in an enterprise application is advisable. Yet, we try to give a short summary: Even in large or intercontinental networks, Fabric can still reach more than 1000 tx/s for public transactions with LevelDB. Using CouchDB and private transactions, each lowers throughput levels by a factor of three lower. The differences become smaller when database operations become less relevant, e.g., for large numbers of endorsers or complex workloads. This means that enterprise consortia can deploy a robust, fault-tolerant system that handles a two- to four-digit number of moderately complex tasks per second, with latencies between a few hundred milliseconds and 3 seconds. While high throughput requirements can only be handled with more expensive hardware, and costs for hardware increase super-linearly in maximum throughput required, the costs for moderate systems can be as low as 10^{-5} USD (provided considerable average utilization of capacity) or a small five-digit USD figure annually even when running on on-demand cloud services. Nonetheless, Fabric reaches its limits for systems requiring a stable throughput of several thousand transactions per second or for systems with slightly lower throughput requirements and an additional need for a large two-digit number of nodes, privacy, or complex workloads. To meet such requirements, further improvements and specific features that allow optimizing the performance of Fabric nodes, such as the ones suggested by Thakkar and Natarajan (2021), will be necessary.

These findings are of great relevance to supply chain management as the proposed blockchain solutions usually have high throughput, low latency, and global infrastructure requirements. Despite covering a comprehensive list of variables, however, the analytic focus of this study required it to leave certain areas unexplored. Future research may, for example, find it beneficial to compare the various supported programming languages, such as Go, Node, and Java. In doing so, we hope that future research will use our extended DLPS framework to examine additional implementations. Furthermore, while this paper considers a recent version of Fabric to evaluate current features, such as private data collections, it is important to note that the blockchain framework is still evolving comparatively fast, with developers focusing on further improving the overall performance along with new features like anonymous credentials and zero-knowledge proofs for improved transaction privacy. Therefore, although our spot-check comparisons between Fabric version 1.4 and 2.0 have indicated that the performance deviations in this update are only small, our analysis is only valid for particular release versions and requires additional testing once an update is introduced. Nonetheless, the list of impact parameters and the functionalities by which we extended the DLPS should be able to support analyses of future updates on the Fabric code. What is more, it supports other enterprise blockchains like Quorum, all of which offer similar functionalities (smart contracts and private transactions) and parameters (such as block time) in deployments with different network architectures and hardware.

Another promising avenue for future research is an examination of the impact of consensus mechanisms on updates of Fabric, and indeed on blockchain implementations in general. Our performance evaluation with RAFT consensus and the comparison with the Solo and Kafka ordering (that are deprecated in version 2.0 but available in 1.4) indicates that the ordering service is not currently a bottleneck in Fabric. However, this may change when a more fault-tolerant consensus mechanism is used, one that involves more communication, such as a three-phase commit in Practical Byzantine Fault Tolerant (PBFT). PBFT was previously used in Fabric version 0.6 but removed in version 1.0, making it impossible to extensively compare this consensus mechanism with others. However, there are ongoing efforts to provide PBFT consensus for Fabric, and since Fabric was designed to be sufficiently modular for various consensus mechanisms to be integrated, several should be available for analysis and comparison in the near future, be they assimilated from other Hyperledger projects (such as Proof of Elapsed Time in Hyperledger Sawtooth, Redundant BFT in Hyperledger Indy) or from other domains entirely (like HoneyBadger BFT or Solana's Proof of History). To this end, it may prove helpful to use the support of DLPS to investigate latency sensitivity with benchmarking systems that are physically far distributed.

A further issue worth bearing in mind when embarking on future research is that, while researchers and practitioners focus on Fabric, we wish to reiterate the need to also investigate the potential of other blockchain implementations. Of course, the architecture of different blockchain systems differs in specific details, but the core architectural principles are common to most. A thorough consideration of architecture, setup, business logic, network, and robustness when benchmarking different blockchain implementations will ultimately facilitate a better comparison of results.

Already, the use and reach of blockchain technology in the industry have come a long way. Especially with the latest releases of Fabric, blockchain implementations have come much closer to operational maturity than ever before. In view of our own research, we are confident that pilot projects will soon be ready for operational implementation, ultimately improving supply chains. Nevertheless, our collective understanding of this technology, its performance, and scalability, in general, remains limited. As a research community, we have only just started to identify a comprehensive set of factors that impact this kind of distributed system. Gaining further insights will refine the design of blockchain infrastructures and applications, pushing the technological boundaries toward more advanced systems. To expedite this development, future research would do well to investigate alternative or complementary approaches to the provision of blockchains that facilitate high performance. This includes pursuing incremental performance optimizations, such as splitting certain workloads to dedicated nodes (Thakkar and Natarajan, 2021), incorporating serverless implementations of nodes or even building completely serverless blockchains that can scale elastically on demand and reach far greater throughput at the tradeoff of increased centralization (Sedlmeir et al., 2022b), or reducing the computational and data complexity of the application itself. The latter could be achieved, for instance, by means of sharding or using succinct verifiable computation techniques, such as zero-knowledge proofs to reduce the workload for multiple nodes at the cost of increased (but heavily parallelizable) computation for one entity (see, e.g., Rückel et al., 2022; Šimunić et al., 2021). At present, succinctly verifiable computation is arguably more relevant for permissionless blockchains because the amortized complexity only pays off when there is a considerable number of validators that need to verify a transaction, but the alternative approaches suggested above may benefit large networks and a high number of endorsers. Our research has led us to believe that by combining these approaches, blockchains can ultimately provide even the scalability required by the most demanding enterprise applications.

Acknowledgment

We thank Philipp Ross, Emil Djerekarov, Jannik Lockl, and Daniel Miehle for their support in creating the first version of the DLPS. Our gratitude further extends to Colin Glass, Marius Poke, and Orestis Papageorgiou for their valuable comments. This research is supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy. We gratefully acknowledge their funding of the project "Fraunhofer Blockchain Center (20-3066-2-6-14)". This research was also funded in part by the Luxembourg National Research Fund (FNR) and PayPal, grant reference "P17/IS/13342933/PayPal-FNR/Chair in DFS/Gilbert Fridgen" (PEARL) and the FiReSpARX Project (C20/IS/14783405/FiReSpARX/Fridgen).

References

Agrawal, T. K., V. Kumar, R. Pal, L. Wang, and Y. Chen (2021). “Blockchain-based framework for supply chain traceability: A case example of textile and clothing industry”. In: *Computers & Industrial Engineering* 154, p. 107130. DOI: 10.1016/j.cie.2021.107130.

Androulaki, E. et al. (2018). “Hyperledger Fabric: A distributed operating system for permissioned blockchains”. In: *Proceedings of the 13th EuroSys Conference*. DOI: 10.1145/3190508.3190538.

AWS (2021). *Amazon EC2 pricing*. Amazon Web Services. URL: https://aws.amazon.com/ec2/pricing/on-demand/?nc1=h_ls.

Azzi, R., R. K. Chamoun, and M. Sokhn (2019). “The power of a blockchain-based supply chain”. In: *Computers & Industrial Engineering* 135, pp. 582–592. DOI: 10.1016/j.cie.2019.06.042.

Baliga, A., N. Solanki, S. Verekar, A. Pednekar, P. Kamat, and S. Chatterjee (2018a). “Performance characterization of Hyperledger Fabric”. In: *Crypto Valley Conference on Blockchain Technology*. IEEE, pp. 65–74. DOI: 10.1109/cvcbt.2018.00013.

Baliga, A., I. Subhod, P. Kamat, and S. Chatterjee (2018b). *Performance evaluation of the Quorum blockchain platform*. URL: http://arxiv.org/abs/1809.03421.

Beck, R., C. Müller-Bloch, and J. L. King (2018). “Governance in the blockchain economy: A framework and research agenda”. In: *Journal of the Association for Information Systems* 19 (10), pp. 1020–1034. DOI: 10.17705/1jais.00518.

Bichsel, P., C. Binding, J. Camenisch, T. Groß, T. Heydt-Benjamin, D. Sommer, and G. Zaverucha (2009). *Cryptographic protocols of the Identity Mixer library*. IBM. URL: http://patrik.biche.ch/pub/rz3730.pdf.

Blockbench repository (2022). URL: https://github.com/ooibc88/blockbench.

Buterin, V. et al. (2014). *A next-generation smart contract and decentralized application platform*. URL: https://github.com/ethereum/wiki/wiki/White-Paper.

Camenisch, J., M. Drijvers, and M. Dubovitskaya (2017). “Practical UC-secure delegatable credentials with attributes and their application to blockchain”. In: *Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security*. ACM, pp. 683–699. DOI: 10.1145/3133956.3134025.

Capocasale, V., D. Gotta, S. Musso, and G. Perboli (2021). “A blockchain, 5G and IoT-based transaction management system for smart logistics: An Hyperledger framework”. In: *45th Annual Computers, Software, and Applications Conference*. IEEE, pp. 1285–1290. DOI: 10.1109/compsac51774.2021.00179.

Casino, F., T. K. Dasaklis, and C. Patsakis (2019). "A systematic literature review of blockchain-based applications: Current status, classification and open issues". In: *Telematics and Informatics* 36, pp. 55–81. DOI: 10.1016/j.tele.2018.11.006.

Chacko, J. A., R. Mayer, and H.-A. Jacobsen (2021). "Why do my blockchain transactions fail? A study of Hyperledger Fabric". In: *Proceedings of the 2021 International Conference on Management of Data*, pp. 221–234. DOI: 10.1145/3448016.3452823.

Chang, S. E. and Y. Chen (2020). "When blockchain meets supply chain: A systematic literature review on current development and potential applications". In: *IEEE Access* 8, pp. 62478–62494. DOI: 10.1109/access.2020.2983601.

Dabbagh, M., M. Kakavand, M. Tahir, and A. Amphawan (2020). "Performance analysis of blockchain platforms: Empirical evaluation of Hyperledger Fabric and Ethereum". In: *2nd International Conference on Artificial Intelligence in Engineering and Technology*. IEEE. DOI: 10.1109/iicaiet49801.2020.9257811.

Dinh, T. T. A., R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang (2018). "Untangling blockchain: A data processing view of blockchain systems". In: *Transactions on Knowledge and Data Engineering* 30 (7), pp. 1366–1385. DOI: 10.1109/tkde.2017.2781227.

Dinh, T. T. A., J. Wang, G. Chen, R. Liu, B. C. Ooi, and K. Tan (2017). *Blockbench: A framework for analyzing private blockchains*. URL: http://arxiv.org/abs/1703.04057.

Dreyer, J., M. Fischer, and R. Tönjes (2020). "Performance analysis of Hyperledger Fabric 2.0 blockchain platform". In: *Proceedings of the Workshop on Cloud Continuum Services for Smart IoT Systems*. ACM, pp. 32–38. DOI: 10.1145/3417310.3431398.

Forbes (2020). *Alibaba's Singles' Day brings in record $74 billion in pandemic year*. URL: https://www.forbes.com/sites/isabeltogoh/2020/11/12/alibabas-singles-day-brings-in-record-74-billion-in-pandemic-year/?sh=6c572495c1b7.

Fraunhofer FIT and Universities of Bayreuth and Luxembourg (2021). *Distributed ledger performance scan repository*. URL: https://github.com/DLPS-Framework.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018). "Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation". In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Geneiatakis, D., Y. Soupionis, G. Steri, I. Kounelis, R. Neisse, and I. Nai-Fovino (2020). "Blockchain performance analysis for supporting cross-border e-government services". In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1310–1322. DOI: 10.1109/tem.2020.2979325.

Guggenberger, T., A. Schweizer, and N. Urbach (2020). “Improving interorganizational information sharing for vendor managed inventory: Toward a decentralized information hub using blockchain technology”. In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1074–1085. DOI: 10.1109/tem.2020.2978628.

Hao, Y., Y. Li, X. Dong, L. Fang, and P. Chen (2018). “Performance analysis of consensus algorithm in private blockchain”. In: *Intelligent Vehicles Symposium*. IEEE, pp. 280–285. DOI: 10.1109/ivs.2018.8500557.

Hyperledger Caliper Repository (2022). URL: https://github.com/hyperledger/caliper.

IBM (2020a). *Blockchain use cases*. URL: https://www.ibm.com/blockchain/use-cases/?bkcsol=platform.

IBM (2020b). *Private data collections on Hyperledger Fabric*. URL: https://github.com/IBM/private-data-collections-on-fabric.

Jensen, T., J. Hedman, and S. Henningsson (2019). “How TradeLens delivers business value with blockchain technology”. In: *MIS Quarterly Executive* 18 (4), pp. 221–243. DOI: 10.17705/2msqe.00018.

Kannengießer, N., S. Lins, T. Dehling, and A. Sunyaev (2020). “Trade-offs between distributed ledger technology characteristics”. In: *ACM Computing Surveys* 53 (2). DOI: 10.1145/3379463.

Kolb, J., M. AbdelBaky, R. H. Katz, and D. E. Culler (2020). “Core concepts, challenges, and future directions in blockchain: A centralized tutorial”. In: *ACM Computing Surveys* 53 (1). DOI: 10.1145/3366370.

Koushik, A., B. Jain, N. Menon, D. Lohia, S. Chaudhari, and V. K. BP (2019). “Performance analysis of blockchain-based medical records management system”. In: *4th International Conference on Recent Trends on Electronics, Information, Communication & Technology*. IEEE, pp. 985–989. DOI: 10.1109/rteict46194.2019.9016812.

Kreps, J., N. Narkhede, J. Rao, et al. (2011). “Kafka: A distributed messaging system for log processing”. In: *Proceedings of the NetDB*. Vol. 11. ACM. URL: https://pages.cs.wisc.edu/~akella/CS744/F17/838-CloudPapers/Kafka.pdf.

Kuzlu, M., M. Pipattanasomporn, L. Gurses, and S. Rahman (2019). “Performance analysis of a Hyperledger Fabric blockchain framework: Throughput, latency and scalability”. In: *International Conference on Blockchain*. IEEE, pp. 536–540. DOI: 10.1109/blockchain.2019.00003.

Labazova, O., T. Dehling, and A. Sunyaev (2019). “From hype to reality: A taxonomy of blockchain applications”. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences*, pp. 4555–4564. DOI: 10.24251/hicss.2019.552.

Lamport, L., R. Shostak, and M. Pease (1982). “The Byzantine generals problem”. In: *ACM Transactions on Programming Languages and Systems* 4 (3), pp. 382–401. DOI: 10.1145/3335772.3335936.

Lim, M. K., Y. Li, C. Wang, and M.-L. Tseng (2021). “A literature review of blockchain technology applications in supply chains: A comprehensive analysis of themes, methodologies and industries”. In: *Computers & Industrial Engineering* 154, p. 107133. DOI: 10.1016/j.cie.2021.107133.

Longo, F., L. Nicoletti, A. Padovano, G. d’Atri, and M. Forte (2019). “Blockchain-enabled supply chain: An experimental study”. In: *Computers & Industrial Engineering* 136, pp. 57–69. DOI: 10.1016/j.cie.2019.07.026.

Ma, C., X. Kong, Q. Lan, and Z. Zhou (2019). “The privacy protection mechanism of Hyperledger Fabric and its application in supply chain finance”. In: *Cybersecurity* 2 (5). DOI: 10.1186/s42400-019-0022-2.

Mattke, J., C. Maier, and A. Hund (2019). “How an enterprise blockchain application in the U.S. pharmaceuticals supply chain is saving lives”. In: *MIS Quarterly Executive* 18 (4), pp. 246–261. DOI: 10.17705/2msqe.00019.

Miehle, D., D. Henze, A. Seitz, A. Luckow, and B. Bruegge (2019). “PartChain: A decentralized traceability application for multi-tier supply chain networks in the automotive industry”. In: *International Conference on Decentralized Applications and Infrastructure*. IEEE, pp. 140–145. DOI: 10.1109/dappcon.2019.00027.

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

Nasir, Q., I. A. Qasse, M. Abu Talib, and A. B. Nassif (2018). “Performance Analysis of Hyperledger Fabric platforms”. In: *Security and Communication Networks*. DOI: 10.1155/2018/3976093.

Nguyen, T. S. L., G. Jourjon, M. Potop-Butucaru, and K. L. Thai (2019). “Impact of network delays on Hyperledger Fabric”. In: *Conference on Computer Communications Workshops*. IEEE, pp. 222–227. DOI: 10.1109/infcomw.2019.8845168.

Ongaro, D. and J. Ousterhout (2014). “In search of an understandable consensus algorithm”. In: *USENIX Annual Technical Conference*, pp. 305–319. URL: https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf.

Perboli, G., V. Capocasale, and D. Gotta (2020). “Blockchain-based transaction management in Smart Logistics: A Sawtooth framework”. In: *Proceedings of the 44th Annual Computers, Software, and Applications Conference*. IEEE, pp. 1713–1718. DOI: 10.1109/compsac48688.2020.000-8.

Perboli, G., S. Musso, and M. Rosano (2018). “Blockchain in logistics and supply chain: A lean approach for designing real-world use cases”. In: *IEEE Access* 6, pp. 62018–62028. DOI: 10.1109/access.2018.2875782.

Pongnumkul, S., C. Siripanpornchana, and S. Thajchayapong (2017). “Performance analysis of private blockchain platforms in varying workloads”. In: *26th International Conference on Computer Communication and Networks*. IEEE. DOI: 10.1109/icccn.2017.8038517.

Reddy, K. R. K., A. Gunasekaran, P. Kalpana, V. R. Sreedharan, and S. A. Kumar (2021). “Developing a blockchain framework for the automotive supply chain: A systematic review”. In: *Computers & Industrial Engineering* 157, p. 1073340. DOI: 10.1016/j.cie.2021.107334.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). “Building a blockchain application that complies with the EU general data protection regulation”. In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Rückel, T., J. Sedlmeir, and P. Hofmann (2022). “Fairness, integrity, and privacy in a scalable blockchain-based federated learning system”. In: *Computer Networks* 202, p. 108621. DOI: j.comnet.2021.108621.

Sedlmeir, J., P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021). “The DLPS: A new framework for benchmarking blockchains”. In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020). “The energy consumption of blockchain technology: Beyond myth”. In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

Sedlmeir, J., J. Lautenschlager, G. Fridgen, and N. Urbach (2022a). “The transparency challenge of blockchain in organizations”. In: *Electronic Markets* 32 (3), pp. 1779–1794. DOI: 10.1007/s12525-022-00536-0.

Sedlmeir, J., T. Wagner, E. Djerekarov, R. Green, J. Klepsch, and S. Rao (2022b). “A serverless distributed ledger for enterprises”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 7382–7391. DOI: 10.24251/hicss.2022.886.

Šimunić, S., D. Bernaca, and K. Lenac (2021). “Verifiable computing applications in blockchain”. In: *IEEE Access* 9, pp. 156729–156745. DOI: 10.1109/access.2021.3129314.

Sunny, J., N. Undralla, and V. M. Pillai (2020). “Supply chain transparency through blockchain-based traceability: An overview with demonstration”. In: *Computers & Industrial Engineering* 150, p. 106895. DOI: 10.1016/j.cie.2020.106895.

Szabo, N. (1997). “Formalizing and securing relationships on public networks”. In: *First Monday* 2 (9). DOI: 10.5210/fm.v2i9.548.

Thakkar, P. and S. Natarajan (2021). “Scaling blockchains using pipelined execution and sparse peers”. In: *Proceedings of the Symposium on Cloud Computing*. ACM, pp. 489–502. DOI: 10.1145/3472883.3486975.

Thakkar, P., S. Nathan, and B. Viswanathan (2018). “Performance benchmarking and optimizing Hyperledger Fabric blockchain platform”. In: *26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, pp. 264–276. DOI: 10.1109/mascots.2018.00034.

Toumia, S. B., C. Berger, and H. P. Reiser (2021). *Evaluating blockchain application requirements and their satisfaction in Hyperledger Fabric*. URL: https://arxiv.org/abs/2111.15399.

Wang, C. and X. Chu (2020). “Performance characterization and bottleneck analysis of Hyperledger Fabric”. In: *Proceedings of the 40th International Conference on Distributed Computing Systems*. IEEE, pp. 1281–1286. DOI: 10.1109/icdcs47774.2020.00165.

Waskom, M. L. (2021). “Seaborn: statistical data visualization”. In: *Journal of Open Source Software* 6 (60).

Xu, X., G. Sun, L. Luo, H. Cao, H. Yu, and A. V. Vasilakos (2021). “Latency performance modeling and analysis for Hyperledger Fabric blockchain network”. In: *Information Processing & Management* 58 (1). DOI: 10.1016/j.ipm.2020.102436.

Zhang, R., R. Xue, and L. Liu (2019). “Security and privacy on blockchain”. In: *ACM Computing Surveys* 52 (3). DOI: 10.1145/3316481.

F Research Paper 5 – A serverless distributed ledger for enterprises

Authors:

Johannes Sedlmeir, Tim Wagner, Emil Djerekarov, Ryan Green, Johannes Klepsch, & Shruthi Rao

Published in:

Proceedings of the 55th Hawaii International Conference on System Sciences
(DOI: 10.24251/hicss.2022.886)

Abstract:

Enterprises have been attracted by the capability of blockchains to provide a single source of truth for workloads that span companies, geographies, and clouds while retaining the independence of each party's IT operations. However, so far production applications have remained rare, stymied by technical limitations of existing blockchain technologies and challenges with their integration into enterprises' IT systems. In this paper, we collect enterprises' requirements on distributed ledgers for data sharing and integration from a technical perspective, argue that they are not sufficiently addressed by available blockchain frameworks, and propose a novel distributed ledger design that is "serverless", i.e., built on cloud-native resources. We evaluate its qualitative and quantitative properties and give evidence that enterprises already heavily reliant on cloud service providers would consider such an approach acceptable, particularly if it offers ease of deployment, low transactional cost structure, and a combination of latency and scalability aligned with real-time IT application needs.

Keywords:

Blockchain engineering, business, cloud, distributed database, DLT, permissioned blockchain

F.1 Introduction

Data, particularly transactional data housed in various flavors of databases, powers the vast majority of modern IT applications. Historically, the bulk of that data was produced and consumed by the owner of the data. However, the growing complexity of supply and logistics chains, the “consumerization” of IT bringing ever-higher expectations for real-time information and automated decision making, and the trend towards simplified software as a service (SaaS) deployments are all causing data to migrate outside a company’s four walls. Classic mechanisms to provide trustworthy, high-fidelity data representation and query results, such as centralized databases offering ACID transactions and SQL query languages, fail when a considerable fraction of data resides elsewhere, accessible only through batch files or by polling third-party APIs, making it potentially inconsistent, incomplete, and out of date. Blockchain technologies appeared to offer a compelling solution to this challenge: A technology that could simultaneously erect a single source of truth in the form of a distributed ledger capable of spanning companies, clouds, and geographical boundaries, while still preserving each individual participant’s control over its own technology stack, including deployment, authentication, security, and compliance needs.

Distributed ledgers used to create distributed, multi-party databases with ACID semantics have a lengthy research history. As far back as the 1980s, researchers investigated crash fault tolerance (CFT) and Byzantine fault tolerance (BFT) state machine replication in order to achieve reliable distributed systems in the presence of failures or adversaries (Lamport et al., 1982). The security of these systems was based on an election mechanism in a *permissioned* setting (two- or three-phase commit), where the identities of all participants or at least their total number was known. Although the first, merely crash-fault tolerant solutions such as Paxos were soon improved, e.g., through Byzantine-fault tolerant protocols such as PBFT (Castro, Liskov, et al., 1999), direct adoption of distributed ledger technologys (DLTs) by enterprises remained rare until recently, although *indirect* usage in the form of public cloud databases that make use of permissioned consensus and Paxos variants became commonplace as cloud adoption has grown (Ailijiang et al., 2016).

The original Bitcoin whitepaper (Nakamoto, 2008) popularized a *permissionless* DLT combined with Sybil attack prevention for the purposes of value storage and transfer that has come to be known as a cryptocurrency. Ethereum expanded on the simple “value transfer" interpreter in Bitcoin with a Turing complete computational engine or smart contract platform (Buterin et al., 2014). Ethereum garnered attention outside the cryptocurrency and speculative financial market communities through its self-marketing as the “world computer". Enterprises and private sector consortia eager for solutions to the inconsistencies, omissions, and high manual reconciliation costs of data silos looked to Ethereum and its variants as a possible solution. At the same time, information systems researchers were attracted by applications of DLT that promised businesses considerable improvements in terms of interoperability, traceability, provenance, distributed control, accountability, and transparency (Beck et al., 2018) by providing a neutral digital infrastructure for cross-organizational processes (Fridgen et al., 2018). Unfortunately, the duplication of computation and storage on every node in the network (Luu et al., 2015) as well as the need for economic incentives imply low throughput, high latency, and significant transaction costs (Kannengießer et al., 2020) and thus make the public permissionless blockchains a non-starter for the vast majority of enterprise use cases, even ignoring potential concerns about exposing their data to the world (Zhang et al., 2019).

Consequently, enterprises have generally found more success with *permissioned* blockchain networks in various sectors, e.g., in improving data exchange and traceability in automotive supply chains (Miehle et al., 2019). Popular open-source implementations of permissioned DLTs include private Ethereum networks such as Quorum, and Hyperledger Fabric (Androulaki et al., 2018). Permissioned blockchains provide many advantages over permissionless blockchains for enterprises, including higher performance, predictable costs and the support of data confidentiality features “off-the-shelf”. Despite these relative advantages, the performance of permissioned blockchains still remains orders of magnitude lower than “centralized” database technologies (Barr, 2019), and – as we will argue in this paper – the costs and complexities associated with setting up and maintaining DLTs for enterprises are significant.

We posit that many of the limitations regarding performance, complexity, and cost in existing enterprise distributed ledger implementations are driven by their reliance on a server-based deployment model and suggest an intriguing alternative: a distributed ledger in which each node is built using “serverless” infrastructure (Castro et al., 2019), thus benefiting from the economic and scaling advantages of massive multi-tenanted implementations that expose inter-machine parallelism opportunities unavailable to prior techniques. Our approach offers the performance and “form fit” of a cloud-based SQL or NoSQL database approach while retaining the decentralized aspect of a permissioned blockchain in the form of segregated accounts containing individually owned resources, in exchange for giving up the ability to run nodes outside of a public cloud setting.

The remainder of this paper is structured as follows: In section F.2 we briefly review serverless architectures and survey related work. Section F.3 derives common enterprise requirements for blockchains used for data integration purposes. In section F.4, we present the main components and characteristics of our serverless blockchain architecture. We then evaluate our implementation from a qualitative and quantitative perspective in sections F.5 and F.6. We summarize our observations and avenues for further research in section F.7.

F.2 Background

F.2.1 Serverless computing

Surveys of serverless offerings and research describe cloud-based compute, storage, queuing, application programming interface (API) hosting, and workflow (choreography) services that offer access to effectively unbounded storage and compute power coupled with a pay-per-call cost structure and latency on the order of 8-10 ms (Jonas et al., 2019). The massively multi-tenanted nature of these services provides an alternative to blockchain algorithms constrained to using a single server per node: Effectively, such a system can "dispatch" thousands of virtual machines in single digit milliseconds, each one verifying or applying an individual transaction within a block. Coupled to the massively parallel front ends of NoSQL databases and blob storage, end-to-end processing and storage parallelism enables individual blockchain nodes to escape the confines of vertical scaling and the prohibitive cost dynamics of scaling each node to peak needs at all times. Reconstructing consensus out of these building blocks exposes multi-machine parallelism opportunities not available in extant blockchain approaches, particularly as conventional consensus algorithms also consider the machine on which they run to be an atomic unit of trust and network identity.

The term "serverless" has entered the lexicon to denote services and architectures that rely on fully managed cloud services (Schleier-Smith et al., 2021). Compared to older application construction methodologies in which companies rent servers from cloud service providers (CSPs) such as Amazon Web Services (AWS), Azure, or Google, serverless architectures rely on the use of services that hide the presence of servers beyond an abstraction layer (Castro et al., 2019). AWS Lambda, a serverless cloud computing service introduced in November 2014, initiated much of the current interest in the category. Lambda works by multi-tenanting both at the fleet and the individual machine level, placing hypervisors around each workload.

Computations are invoked by HTTPS requests and routed to a (possibly preexisting) container by a low single-digit millisecond bin packing router that is tenant aware. Cloud function services from other CSPs work similarly. Serverless CSP services, and the applications constructed using them, are typically differentiated along several dimensions of interest to our analysis:

- *Intrinsically fault tolerant:* The "gold standard" for a highly available (99.9 – 99.99 % uptime) system is 3-way redundancy across spatially isolated data centers – what AWS refers to as availability zones (AZs). Serverless offerings hence build fault tolerance into their implementation, so the service *as delivered to the application* includes redundancy by design. By contrast, each participant in a conventional blockchain would need to own and operate at least three nodes themselves, just to ensure an equivalent availability outcome.
- *Scale-per-request:* With a conventional architecture, scaling up the operational capability of a system requires either vertical or horizontal scaling techniques; i.e., either "rent a bigger box" or "rent more boxes". Serverless services manage that scaling behind the scenes, typically relying on massively multi-tenanted fleets, which provides the illusion of essentially limitless scaling driven exclusively through making requests to the service's API.
- *Pay-per-request:* Serverless offerings typically charge on a per-request basis (rather than a time-based rental fee), and thus unlike infrastructure-based architectures they "turn off" completely, generating no costs when not in use (Castro et al., 2019). Given that typical enterprise fleets only achieve around 18 % utilization (451 Research, 2019), this can represent a significant improvement in costs and also energy consumption.

F.2.2 Related work

Researchers have already started to analyze the tradeoffs and challenges that come with blockchain adoption from a technical (Kannengießer et al., 2020) and organizational perspective (Zavolokina et al., 2020). Moreover, work like Lacity (2018) focused on structuring standardization, performance, and regulatory requirements and developed strategies to address associated challenges. Recently, publications have analyzed the business-related challenges in specific application areas, such as supply chains (Hastig and Sodhi, 2020). Chatterjee et al. (2019) discusses challenges of DLT adoption from the perspective of enterprises from a review of literature based on a weighted average score. However, to our knowledge, so far no large-scale study involving enterprises has been conducted to determine enterprises' requirements on integrating distributed ledger technologies in their IT infrastructure.

To locate additional related work that aims to propose a serverless blockchain design, we applied the search term "serverless AND (blockchain OR distributed ledger)" in the arXiv, ACM digital library, Google scholar, and IEEE Xplore databases. We found that Oh and Kim (2019) suggested an architecture in which some computational tasks for the Hyperledger Sawtooth blockchain can be shifted to AWS Lambda. Beyond this, Ghaemi et al. (2020) uses a serverless approach in another sense, namely shifting computational work in the context of a blockchain to a user's personal devices. Finally, Kaplunovich et al. (2019) use AWS Lambda for the client side, sending requests to a Fabric network for a performance analysis. Recently, a cryptographically verifiable SQL Database for Azure has been proposed (Antonopoulos et al., 2021) that has some similarities with the storage layer of the serverless blockchain that we propose; however, this service is restricted to a single account and hence does not explore consensus-related topics that could support the synchronization of data across multiple accounts. Consequently, we found no work that suggests an architecture for or creates a fully functional distributed ledger based on serverless components.

F.3 Business requirements for blockchains

To collect requirements for enterprise blockchains for data sharing and integration from practitioners, we interviewed 1,092 companies having at least some prior public cloud experience in person from 2017 through 2019. They spanned enterprises, SMBs, and startups and represent verticals such as automotive, financial services, consumer packaged goods, food & beverage, travel & hospitality, media & entertainment, agriculture, IT, telecom and semiconductors, and public sector. Over 95 % purchased services from AWS and over 91 % spent at least $50,000 per month on cloud services. 98 % of interviewed companies were for-profit and nearly two thirds were enterprises. All interviews were 60 minutes or more in duration and included both structured feedback and free-form inquiry regarding data sharing and application construction requirements, intended use cases for blockchain or ledger-like offerings, and – where applicable – reasons for adopt-

Capability	**Requirements**
Decentralization	Each participant must be able to maintain a legally and operationally independent copy of all data and metadata without reliance on another company's IT organization.
(Multi-) cloud deployment	DLTs used for data integration purposes must be deployable to public clouds in order to integrate with existing IT security and operations. Each node must also be able to make an independent decision with respect to the choice of CSP, enabling participants to achieve low-latency interconnect with other resources and services in that CSP.
Elastic scaling on demand	The DLT must support flexible and effectively instantaneous scalability to accommodate enterprise IT workloads, which may vary unpredictably. A cost structure that scales linearly (versus being scaled perpetually to peak capacity) is a positive differentiator.
Unlimited storage	Enterprises expect DLTs used as data storage and integration solutions to operate without limits on any form of storage (file, blob, database size, etc.).
Fault tolerance and high availability	99.99 % availability is the standard for enterprise contracts, with mission critical and financial systems often requiring 99.999 % uptime. Having the DLT provide this capability intrinsically with no additional cost, deployment complexity, or maintenance on the part of the user is a positive differentiator.
Ease of deployment	Conventional blockchain deployments often demand non-trivial staffing to configure, deploy, and maintain. Schema-driven definition (similar to conventional database tables), SaaS-based delivery, and limiting manual labor required for networking, operating system, virtual machine, security, or availability configuration are positive differentiators.
Low latency and fast finality	Many online transaction processing (OLTP) tasks in enterprise applications have near real-time data processing expectations, requiring fast (sub-second) confirmation of transactions.
Energy efficiency	The DLT's effective utilization, and its reliance on the power grid, must be in line with typical corporate applications. Improved utilization relative to state of the practice is a differentiator.
Access control and data governance	Nearly all enterprises require the ability to scope ledger and world state updates to a subset of participants on a per-transaction basis ("private transactions"), for reasons ranging from business confidentiality, to material disclosure laws, to data protection regulations.

Table 1: Business requirements for DLTs used for enterprise data sharing and integration applications.

ing or abandoning blockchain technology. We did not select interviewees specifically for success or failure of DLT projects, but all of our interviewees had expressed interest in, or were actively involved with, a DLT project.

The most frequently cited reason for adopting (or intending to adopt, as was more often the case) blockchain technology was what we termed "dispersed data" problems: Internal data that spanned departments and/or multi-company workflows that spanned business partners, such as suppliers or logistics. Frequently, this data also had to traverse at least one other divide: multiple geographies, multiple providers (AWS, Azure, Snowflake, and Databricks were the most frequently cited), or needed to straddle an on-premise/public cloud connection. Interviewees often chose terms such as, "single source of truth", "shared system of record", "breaking down data silos", "connecting data", or "multi-party solutions" to express their desired end states and their reason for considering blockchain as a solution. Highly correlated requirements included privacy and security concerns, with interviewees often stressing that some form of access controls were mandatory to enable them to "keep control of their data", and the public blockchains were thus often a non starter as a place to store actual business data. Secondary concerns included deployment and operating costs, educational costs (e.g., specialized languages, training, or access to distributed systems and blockchain experts), and ease of partner onboarding and offboarding.

Unsurprisingly, given that the interviewees were selected for their interest in public cloud technologies, none of the respondents considered the ability to run a blockchain solution "on premise" a requirement; in fact, the overwhelming majority requested fully managed solutions, using terms such as "SaaS-style". These requests were frequently coupled with concerns over operational and staffing complexity, with most interviewees acknowledging that, despite their interest, they were unprepared to staff, develop, or operate either public or private blockchain infrastructure at the time of interview. Another non-requirement we discovered was tokenization – most interviewees agreed with the approach taken by Hyperledger Fabric and other permissioned solutions where nodes are treated as conventional enterprise infrastructure costs and there is no economic incentive desired or required for operation. Below we present two anecdotal but typical quotes gleaned from interviewees, and summarize the highest voted results from asking interviewees to select their top 5 requirements for distributed ledger technologies in Table 1.

CEO of a Leading Airline Alliance: *"To ensure appropriate and timely responses to market changes, businesses need to be highly agile, ensure connected experiences and tie cost to demand. We are a highly connected and complex industry and we succeed at delivering the best outcome to the travelers only if all partners are able to make decisions on a single, agreed upon version of the truth. Managing point solutions is expensive and the fixed costs are high and neither scalable nor agile. What we need is a highly scalable and agile multi-lateral agreement mechanism with a SaaS-like model."*

CEO of a Leading Insurance Provider: *"We need all the promises of Blockchain – a single source of truth with each party controlling their own data – but with the scale, cost advantages, and enterprise-grade feature set of a public cloud service."*

Of the approximately 27% of interviewees already engaged in any form of DLT deployment (from prototyping through production attempts), the overwhelming majority reported a lack of success or significant impediments. This includes nearly 100% churn among public-Ethereum-based trials and a striking 90% abandonment rate for active

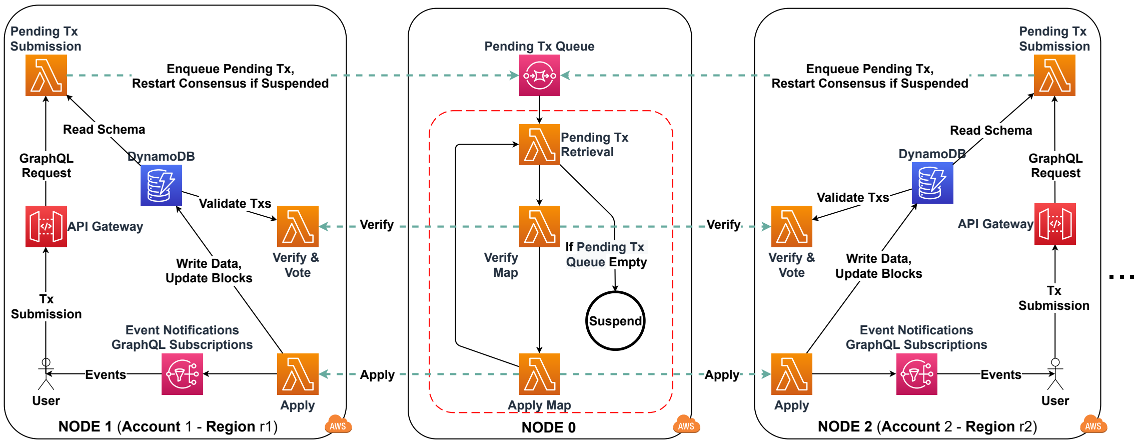


Figure 1: Components and transaction lifecycle of a serverless blockchain.

PoCs, pilots, or other trials involving Hyperledger Fabric, with the remainder either incomplete at time of discussion or scoring poorly on likelihood of eventual implementation. The most frequently cited reasons for terminating a project were costs and complexity, with PoCs typically requiring 6-12 months and infrastructure and staffing or consulting costs that in many cases exceeded $1M USD. Attempts to simulate partner onboarding registered the highest levels of complaints and failures, due to the additional costs, deployment, and connectivity burdens involved, which sometimes even led to scenarios in which one of the parties ran all the nodes in the blockchain network – a setting that contradicts the original intention of a DLT.

SOC2, GDPR, PCI, and other compliance programs that address regulation were typically cited as requirements, and interviewees also expressed *de jeure* concerns: For example, the climate impact of proof-of-work (PoW) solutions and its well-known energy consumption (Sedlmeir et al., 2020) often failed to meet shareholder and customer expectations regarding a public corporation's environmental impact.

F.4 A serverless DLT architecture

Out of the box, Serverless cloud services share a key limitation with the earlier cloud technologies: A centralized, single-owner resource model. Moreover, these resources are mutable by the owner. Using them to construct a blockchain consensus algorithm and, thus, a multi-party, decentralized ledger requires algorithmic techniques that differ from both conventional consensus approaches and classic "single party" cloud application design. Figure 1 illustrates the high-level architecture of the core of a serverless blockchain, analogous to the pending transaction ingestion, data replication and durability aspects, and consensus ("block minting") elements of a conventional blockchain. Transactions follow a distributed two-phase commit lifecycle:

(1) An API Gateway receives the transaction as an HTTPS request, and uses a serverless function (AWS Lambda in our implementation) to add it to a (durable) shared pending transaction queue. Users can group their updates into ordered or unordered atomic transactions. (2) A serverless choreography service reads pending transactions from the queue, combines them into a batch ("block") by applying commutativity and associativity proofs that will enable non-deterministic parallelism, then orchestrates a two-phase commit among all nodes again using serverless functions. This "leader" function can be rotated amongst the nodes or operated in a separate account from them. (3) In the first ("verify") phase, each node checks the syntactic and semantic validity of transactions; the pending block is also written to durable storage. (4) In the second ("apply") phase, transactions are committed to world state and the block is marked committed. In both phases, transactions within a block are processed in parallel, a key performance difference with prior approaches. The ledger is a typical blockchain-style data structure in which each record contains a hash of its own content as well as a hash pointer to the previous block's content to provide tamper-evidence. Signed hashes from each of the nodes participating in the block's construction can be included to make the ledger a standalone "proof of agreement" that can be independently audited. Both ledger and world state are stored in a cloud-based NoSQL data store.

Conventional distributed application techniques can be incorporated into the algorithm above to enable individual nodes to fail (i.e., either verify or apply calls are not returned) and to re-synchronize them to the group, providing fault tolerance up to whatever degree (majority, Byzantine Attack-resistant majority, etc.) the chain's policy permits.

A naïve implementation of the above sketch would require centralized trust: A nefarious orchestration, e.g., could ignore the actual votes from the first commit phase. To explore the construction of a decentralized approach, we define a simplified threat model we refer to as *downward-only trust with identities*: Parties in the chain do not trust each other, but can reliably identify messages (either through the use of CSP identity mechanisms or any form of key pairs). As with conventional blockchains, all parties trust their "infrastructure" – e.g., assume that the CPU processors, data centers, and so forth faithfully execute the consensus algorithm as written. Network messages are assumed to be subject to loss and/or corruption by an adversary. Because our approach is cloud-based, denial of service (DOS) attacks are trivial to reject at the infrastructure level and do not appear in the consensus algorithm per se.

To counter threats, we rely on a combination of techniques to convert “centralized”, single-owner computations into multi-party ones. Chief among these is *verifiable immutability*: By rendering a resource immutable to everyone (including its creator) in a way that can be independently verified by others, trust in its content switches from an (untrusted) fellow participant in the chain to the (trusted) transitive closure of infrastructure. In conjunction with consensus, these techniques enable “party-independent” storage and compute along with the ability to verify both consensus correctness and enforce application-defined smart contract code reviews with effectively no performance overhead, i.e., $O(1)$ time relative to transaction submission and block construction.

Code Storage: Reference copies of code used to perform consensus and for user-provided smart contracts can be rendered immutable through the embargo features provided by all major CSP blob storage services.

Compute: The versioning of cloud functions enables immutable execution, where the outcome is provably independent of the identity of both the owner and the caller. In addition, we rely on the ability of CSPs to provide either the code or a hash for a function’s content in a reliable way.

Orchestrations: We utilize CSP immutability and/or versioning features to acquire a read-only copy of the orchestration that is guaranteed to be linked to an in-flight execution of same, and then prove its correctness by having verifiers vote on its veracity.

The serverless system that we developed includes a compiler capable of converting a JSON Schema-based representation of a data model into the multi-party, cloud-hosted deployment described above. Data integrity is handled as in prior approaches: hash chaining and signatures from all verifiers voting “yes” that include the block’s id and hash protect against future attempts to corrupt, reorder, or repudiate transaction content and enable automated correctness proofs for materialized world state at any block height. By including software updates, metadata correctness proofs, smart contract code agreements, and data schema evolution as block entries, the trust model can be naturally extended to correctness proofs of the consensus algorithm itself (including software patches) and contract execution. Including transaction-submitted hashes and signatures extends this approach to protecting the individual content prior to submission.

F.5 Qualitative evaluation

We now discuss our approach to implementing a permissioned blockchain from a qualitative perspective, structured according to the business requirements collected in section F.3.

While our approach allows for individual nodes to be placed on the owner's preferred CSPs, conventional ("server-based") blockchains also permit operating an individual node outside of any cloud, for example in a self-hosted data center. Recreating the fault tolerance inherent in our approach, would of course entail additional costs in that model to create multi-region data centers with decorrelated fault models. In choosing a cloud-native implementation, our approach restricts the set of providers to public cloud CSPs; our interview results indicated that companies are already reliant on or more CSPs for critical business processes and thus find this acceptable and in many cases preferable, as it simplifies deployment, management, hosting, and administration for them. The critical locus of trust for our survey respondents was with respect to other business parties participating in the chain, rather than whether the chain itself is hosted in the cloud or on premise, and they were comfortable trusting a provider such as AWS, in much the same way they trust a company like Intel at the processor level. Consequently, our serverless distributed ledger approach can sufficiently address enterprises' **decentralization** needs.

Existing permissioned blockchains like Fabric or Quorum are known to be both CPU bound and to expose limited multi-CPU/multi-core parallelism, restricting their ability to scale elastically on demand (Sedlmeir et al., 2021). Popular permissioned blockchains like Fabric and Quorum also employ databases, such as LevelDB or CouchDB, that have storage limits, unlike our approach's reliance on (the effectively limitless) cloud-hosted NoSQL database storage engines. According to several interviewees, LevelDB and CouchDB are uncommon in enterprise IT stacks, and having enterprise security teams authorize them can impose time-consuming analysis. Consequently, integrating a blockchain such as Hyperledger Fabric into a modern IT stack can require substantial infrastructure work, including networking, server allocation and maintenance, and long-term data storage considerations. By contrast, our serverless blockchain approach is natively compatible with common cloud-based storage solutions, and the inherently multi-tenanted infrastructure and economies of scale enable our algorithm to exploit massively parallel data writing bandwidth to both NoSQL and blob storage services from within the compute layer. As a result, our ledgers, world state, and on-chain blob storage are all effectively unlimited – CSPs simply grow their underlying physical data centers over time. Consequently, a serverless blockchain implementation addresses enterprise requirements regarding **elastic scaling on demand** and **unlimited storage** by design.

A further benefit of using serverless technologies is that they natively incorporate **fault tolerance and high availability** into their implementations, relieving their owners of the responsibility of constructing and managing the associated scaling and monitoring infrastructure. This is also a cost optimization, as both the human and infrastructure costs of scaling and operating large fleets and then packing work into them is amortized across millions of users with heterogeneous loads. By contrast, a server-based DLT must deploy multiple nodes (in the case of AWS, e.g., typically three nodes in three AZs) to achieve a 99.99 % availability service level agreement (SLA), and scale vertically to peak load requirements. Furthermore, while server-based nodes can crash, particularly under high load, the intrinsic fault tolerance of serverless computing methods admits to a distributed ledger design in which a single transaction or block might fail, but the system as a whole remains resilient, particularly as each resource (serverless function, orchestration, storage unit, etc.) offers fault tolerance independent of other components. By contrast, a node in a server-based blockchain implementation generally fails as a unit.

By design, a serverless distributed ledger addresses **cloud deployment** requirements, and features inherent to serverless resources also improve the **ease of deployment**: Measuring time-to-market objectively is a difficult exercise, but qualitatively a serverless approach is far simpler than one that exposes the details of server-based networking and infrastructure. In concrete terms, a serverless model allows not only the seamless integration with cloud-based services and legacy systems but also the reuse of well established building blocks, in particular, CSP key distribution, identity and access management (IAM), and production-grade security. In our approach, a multi-party, multi-region, multi-CSP production solution can be constructed and deployed from a data model in under 10 minutes, even up to hundreds of participants; similar approaches for highly available server-based blockchains, even when performed by highly experienced teams operating with large personnel and hardware budgets, would typically be in the range of weeks to multiple quarters, based on feedback from the interviews described in section F.3. Our approach also supports fully managed (aka “SaaS”) deployments, in which the accounts and resources associated with a given node are constructed by our system on behalf of that participant.

Extant literature has already demonstrated that permissioned blockchains like Fabric and Quorum generally exhibit **low latency and fast finality** on the order of several hundreds of milliseconds to a few seconds (Sedlmeir et al., 2021) and hence significantly improve on their public blockchain counterparts. While this is already suitable to address many enterprises' requirements, a serverless approach can further improve on these outcomes through the use of massively parallel computation and – at least for intra-datacenter applications – access to CSP dark fiber. Moreover, permissioned blockchains' **energy consumption** is orders of magnitude below that of PoW-cryptocurrencies, determined by the number of nodes as this reflects the degree of redundancy for the operation and storage of transactions (Sedlmeir et al., 2020). Our approach improves further on this result by collapsing energy consumption and costs to be linear in transaction processing, rather than a function of continuously operated peak capacity; this is made possible by employing a highly multi-tenanted substrate that can effectively share compute, storage, and network capacity across many users with spatially and temporally decorrelated workloads.

Privacy and **access control**, especially the ability of a transaction's submitter to subset the viewers or updaters of its content amongst chain participants, is a critical enterprise feature. For example, Quorum and Hyperledger Fabric support private transactions that are only stored and executed in non-obfuscated form by the intended recipients (Androulaki et al., 2018). We have extended our verifiable immutability approach to include enforcement of policies, using it to create access control lists (ACLs) on all fields, regardless of size or data type. ACLs are themselves stored in the ledger, enabling full auditing and lineage tracking for permission-based metadata in the same way that the underlying data itself is managed and queried. While this approach yields essentially the same functionality as private transactions offered by some existing permissioned blockchains, it considerably simplifies the ease of deployment, for example, compared to Quorum where the additional setup of a software-based enclave is necessary, or Fabric, where access control lists need to be specified individually for every smart contract at the time of deployment.

To further substantiate our qualitative arguments, we also selected a subset of 50 companies previously interviewed and presented our vision of a serverless distributed ledger. 45 of those indicated that the approach was 'likely' or 'very likely' to meet their needs, and 10 companies have already piloted or deployed a commercialization of this approach, half of which used it to replace Ethereum or Hyperledger.

F.6 Quantitative evaluation

To examine the **performance** effect of multi-machine parallelism and access to massive data transfer parallelism available in the public cloud, we compared our approach to two permissioned blockchains, Fabric and Quorum, as these generally exhibited the best performance among several permissioned blockchains in a performance analysis (Sedlmeir et al., 2021). For our benchmarks, we leveraged the distributed ledger performance scan (DLPS), a standardized tool for determining maximum throughput, latency, and resource metrics (Sedlmeir et al., 2021). We set up a Fabric network with 8 peers and 4 orderers, and a Quorum network of 8 nodes to compare with an 8-“node” serverless blockchain. We investigated different choices of hardware in AWS for the server-based blockchains, and chose a simple transaction payload in all cases (writing a single key-value pair). All benchmarks were conducted with default user account settings in AWS, with no limit increases. We tested single-datacenter deployments, cross-European deployments with two datacenters in Frankfurt and Dublin, and an intercontinental setup with four datacenters in Singapore, Sao Paolo, Frankfurt, and Virginia to explore geo-related latency sensitivities. Even for very expensive hardware (16 vCPUs per node), the maximum throughput of Fabric and Quorum did not exceed 4,000 tx/s (compare also Androulaki et al. (2018), Baliga et al. (2018), and Thakkar and Natarajan (2021)), with latencies of around 1-2 seconds. Significantly, *a request rate in excess of the maximum throughput frequently leads to the crash of at least one node within less than a minute* (see also the discussion of local fault tolerance in section F.5), requiring manual intervention to recover. By contrast, the serverless blockchain achieved a maximum ingress rate of more than 8,000 tx/s in all scenarios (up to 75,000 tx/s in the single datacenter scenario), and remained resilient well beyond this rate. Our initial prototype, without commit-time parallelism, was limited to 200 tx/s due to its use of an off-the-shelf cloud orchestration service. Preliminary results from rewriting our consensus in the form of cloud functions indicate that we can effectively parallelize thousands of world state updates per block, effectively exploiting the massively parallel data-planes available in cloud-based NoSQL data stores to match ingestion rates. We consequently expect that we can reach a commit throughput of several thousands of transactions per second in an optimized version, and more than 10,000 transactions per second with customized CSP account settings.

For a server-based blockchain, the operating **cost** per transaction related to infrastructure is straightforward to compute: Assuming the same hardware for all nodes, the costs per second are simply the costs for all servers per second. Low throughput, thus, means that costs per transaction are high. When attempted throughput approaches maximum throughput, the costs for a server-based blockchain can become very low; however, our results (see above) suggest these systems become increasingly unreliable when actual loads near maximum capacity. In contrast, owing to the multiple components that are invoked and billed separately during the lifecycle of a transaction, the cost structure for a serverless blockchain is considerably more complex. In general, the cost structure in

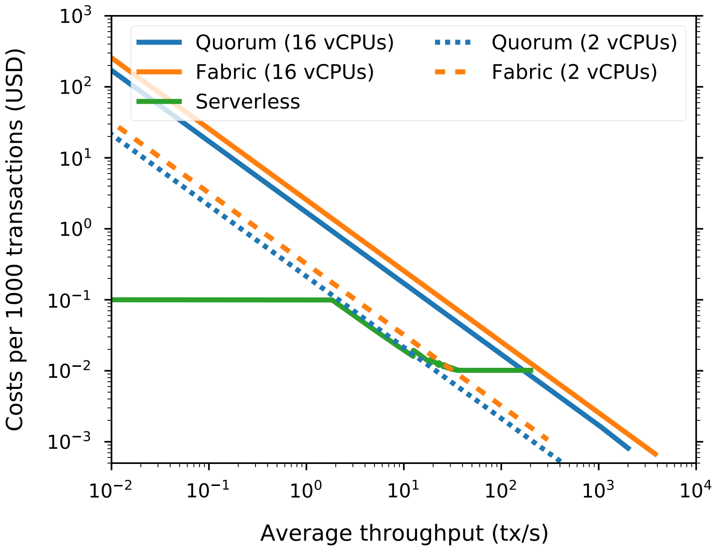


Figure 2: Comparison of per-transaction costs for serverful and serverless distributed ledgers.

our approach involved both a fixed overhead per transaction and both fixed and variable per-block overhead. For small request frequencies, block sizes will typically have a single transaction ($n = 1$), and experimentally, we determined a cost of $ 0.0001; at the maximum packing size $n \approx 900$, of our tested implementation, amortizing block costs reduced this to $ 0.00001 on a per transaction basis. Unsurprisingly, compute costs (cloud function invocations) dominated our cost structure in this experimental setup, yet larger payload sizes could alter that in favor of data transfer or storage costs. An interesting characteristic of our approach is that it allows clients to express their latency sensitivity: While a transaction that needs low latency cannot be more expensive than the costs for $n = 1$, specifying a high latency bound could enable lower transaction prices in the presence of infrequent arrival rates by allowing larger blocks to be minted.

Figure 2 compares per-transaction costs for Fabric, Quorum and our serverless solution based on average throughput. The costs of our serverless implementation have caps due to the minimum and maximum batch size and interpolate cost at intermediate batch sizes. For infrequent arrival rates, serverless significantly outperforms server-based solutions; conversely, server-based solutions shine when they are operated just below their point of failure. Many corporate application workloads are not constant, of course, and as volatility increases, the ratio of maximum throughput to average throughput increases. For server-based blockchains, this requires more expensive hardware (in the form of vertical scaling), increasing per-transaction costs, whereas serverless solutions approximate costs that are linear in the number of transactions processed regardless of scale or volatility. Finally, for this comparison we did *not* create the 3-way redundancy required to achieve an approximately equivalent level of fault tolerance on Fabric or Quorum; multiplying the costs of those systems by 3X would yield a comparable outcome in this regard, strongly favoring the serverless approach.

F.7 Conclusions and future research

In this paper, we collect enterprise requirements for blockchains to enable cross-organization data exchange and propose an approach that combines many of the decentralization benefits of conventional distributed ledger approaches with the advantages of multi-tenanted but centralized cloud services. While blockchains may be employed for a wide variety of purposes, our approach aligns with the needs of business users attempting to construct a "single source of truth" among untrusted business parties. Our contributions include exploration of blended approaches that lie neither in centralized nor conventional ("server-based") decentralized algorithms, and which are capable of exploiting massive multi-machine parallelism to overcome scaling and smart contract processing limitations inherent in single-box approaches, while still exhibiting useful decentralized outcomes such as isolation and consistent data replication among nodes. Benchmarking of two popular permissioned blockchains, Fabric and Quorum, against our serverless implementation in terms of throughput and costs indicates that our current implementation already improves on multiple performance aspects – including transaction ingress rates well in excess of those achievable through conventional means, while further optimizations promise to outperform existing permissioned blockchains through readily exploited avenues, such as decoupling of transaction content copying from consensus. Future work will also focus on establishing lower bounds for blockchains in which compute, storage, and network capacity are effectively unbounded, such as highly parallelizeable associativity and commutativity proofs, and a quantitative study of smart contract performance comparisons to conventional approaches. We also aim to rigorously evaluate whether the projects that leverage our serverless distributed ledger will have a higher success rate than what we found for existing permissioned blockchains in our interview study.

Our initial results were produced on AWS, and some of the features on which we relied, such as function versioning, are not fully implemented on other providers, requiring additional or alternative approaches. More interesting as a research avenue is cross-cloud fault tolerance, in which consensus can span CSPs and survive temporary outages in much the same way that the existing system can survive regional outages *within* a CSP. We hypothesize that selective use of conventional consensus algorithms across clouds could be applied in such a way as to minimize the performance impact while offering enhanced threat models and availability guarantees and hope to explore such patterns in future work.

References

451 Research (2019). *The carbon reduction opportunity of moving to Amazon Web Services*. URL: https://shorturl.at/ajmtK.

Ailijiang, A., A. Charapko, and M. Demirbas (2016). “Consensus in the cloud: Paxos systems demystified”. In: *25th International Conference on Computer Communication and Networks*. DOI: 10.1109/icccn.2016.7568499.

Androulaki, E. et al. (2018). “Hyperledger Fabric: A distributed operating system for permissioned blockchains”. In: *Proceedings of the 13th EuroSys Conference*. DOI: 10.1145/3190508.3190538.

Antonopoulos, P., R. Kaushik, H. Kodavalla, S. R. Aceves, R. Wong, J. Anderson, and J. Szymaszek (2021). “SQL Ledger: Cryptographically verifiable data in Azure SQL database”. In: *Proceedings of the International Conference on Management of Data*. ACM. DOI: 10.1145/3448016.3457558.

Baliga, A., I. Subhod, P. Kamat, and S. Chatterjee (2018). *Performance evaluation of the Quorum blockchain platform*. URL: http://arxiv.org/abs/1809.03421.

Barr, J. (2019). *Amazon Prime day 2019 – Powered by AWS*. URL: https://aws.amazon.com/blogs/aws/amazon-prime-day-2019-powered-by-aws/.

Beck, R., C. Müller-Bloch, and J. L. King (2018). “Governance in the blockchain economy: A framework and research agenda”. In: *Journal of the Association for Information Systems* 19 (10), pp. 1020–1034. DOI: 10.17705/1jais.00518.

Buterin, V. et al. (2014). *A next-generation smart contract and decentralized application platform*. URL: https://github.com/ethereum/wiki/wiki/White-Paper.

Castro, M., B. Liskov, et al. (1999). “Practical Byzantine fault tolerance”. In: *Proceedings of the 3rd Symposium on Operating Systems Design and Implementation*, pp. 173–186. URL: https://pmg.csail.mit.edu/papers/osdi99.pdf.

Castro, P., V. Ishakian, V. Muthusamy, and A. Slominski (2019). “The rise of serverless computing”. In: *Communications of the ACM* 62 (12), pp. 44–54. DOI: 10.1145/3368454.

Chatterjee, A., M. Parmar, and Y. Pitroda (2019). “Production challenges of distributed ledger technology (DLT) based enterprise applications”. In: *International Symposium on Systems Engineering*. DOI: 10.1109/isse46696.2019.8984533.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018). “Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation”. In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Ghaemi, S., H. Khazaei, and P. Musilek (2020). “ChainFaaS: An open blockchain-based serverless platform”. In: *IEEE Access* 8, pp. 131760–131778. DOI: 10.1109/access.2020.3010119.

Hastig, G. M. and M. S. Sodhi (2020). “Blockchain for supply chain traceability: Business requirements and critical success factors”. In: *Production and Operations Management* 29 (4), pp. 935–954. DOI: 10.1111/poms.13147.

Jonas, E., J. Schleier-Smith, V. Sreekanti, C.-C. Tsai, A. Khandelwal, Q. Pu, V. Shankar, J. Carreira, K. Krauth, N. Yadwadkar, J. E. Gonzalez, R. A. Popa, I. Stoica, and D. A. Patterson (2019). *Cloud programming simplified: A Berkeley view on serverless computing*. URL: https://arxiv.org/pdf/1902.03383.

Kannengießer, N., S. Lins, T. Dehling, and A. Sunyaev (2020). “Trade-offs between distributed ledger technology characteristics”. In: *ACM Computing Surveys* 53 (2). DOI: 10.1145/3379463.

Kaplunovich, A., K. P. Joshi, and Y. Yesha (2019). “Scalability analysis of blockchain on a serverless cloud”. In: *IEEE International Conference on Big Data*, pp. 4214–4222. DOI: 10.1109/bigdata47090.2019.9005529.

Lacity, M. C. (2018). “Addressing key challenges to making enterprise blockchain applications a reality”. In: *MIS Quarterly Executive* 17 (3), pp. 201–222. URL: https://aisel.aisnet.org/misqe/vol17/iss3/3.

Lamport, L., R. Shostak, and M. Pease (1982). “The Byzantine generals problem”. In: *ACM Transactions on Programming Languages and Systems* 4 (3), pp. 382–401. DOI: 10.1145/3335772.3335936.

Luu, L., J. Teutsch, R. Kulkarni, and P. Saxena (2015). “Demystifying incentives in the consensus computer”. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 706–719. DOI: 10.1145/2810103.2813659.

Miehle, D., D. Henze, A. Seitz, A. Luckow, and B. Bruegge (2019). “PartChain: A decentralized traceability application for multi-tier supply chain networks in the automotive industry”. In: *International Conference on Decentralized Applications and Infrastructure*. IEEE, pp. 140–145. DOI: 10.1109/dappcon.2019.00027.

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

Oh, B. and D. Kim (2019). “Serverless-enabled permissioned blockchain for elastic transaction processing”. In: *Proceedings of the 20th International Middleware Conference Demos and Posters*, pp. 9–10. DOI: 10.1145/3366627.3368118.

Schleier-Smith, J., V. Sreekanti, A. Khandelwal, J. Carreira, N. J. Yadwadkar, R. A. Popa, J. E. Gonzalez, I. Stoica, and D. A. Patterson (2021). “What serverless computing is and should become: The next phase of cloud computing”. In: *Communications of the ACM* 64 (5), pp. 76–84. DOI: 10.1145/3406011.

Sedlmeir, J., P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021). “The DLPS: A new framework for benchmarking blockchains”. In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020). “The energy consumption of blockchain technology: Beyond myth”. In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

Thakkar, P. and S. Natarajan (2021). “Scaling blockchains using pipelined execution and sparse peers”. In: *Proceedings of the Symposium on Cloud Computing*. ACM, pp. 489–502. DOI: 10.1145/3472883.3486975.

Zavolokina, L., R. Ziolkowski, I. Bauer, and G. Schwabe (2020). “Management, governance and value creation in a blockchain consortium”. In: *MIS Quarterly Executive* 19 (1). DOI: 10.17705/2msqe.00022.

Zhang, R., R. Xue, and L. Liu (2019). “Security and privacy on blockchain”. In: *ACM Computing Surveys* 52 (3). DOI: 10.1145/3316481.

G Research Paper 6 – The transparency challenge of blockchain in organizations

Authors:

Johannes Sedlmeir, Jonathan Lautenschlager, Gilbert Fridgen, & Nils Urbach

Published in:

Electronic Markets (DOI: https://doi.org/10.1007/s12525-022-00536-0)

Abstract:

This position paper discusses the challenges of blockchain applications in businesses and the public sector related to an excessive degree of transparency. We first point out the types of sensitive data involved in different patterns of blockchain use cases. We then argue that the implications of blockchains' information exposure caused by replicated transaction storage and execution go well beyond the often-mentioned conflicts with the GDPR's "right to be forgotten" and may be more problematic than anticipated. In particular, we illustrate the trade-off between protecting sensitive information and increasing process efficiency through smart contracts. We also explore to which extent permissioned blockchains and novel applications of cryptographic technologies such as self-sovereign identities and zero-knowledge proofs can help overcome the transparency challenge and thus act as catalysts for blockchain adoption and diffusion in organizations.

Keywords:

Confidentiality, data protection, digital wallet, distributed ledger technology, privacy, verifiable computation

G.1 Introduction

In the past decade, Bitcoin, Ethereum, and other cryptocurrencies have swiftly made their way from a few cypherpunks' revolutionary vision to a now almost mainstream family of financial assets and decentralized applications. For instance, the investment bank Morgan Stanley recently announced that it now offers their wealthy clients Bitcoin or other crypto exposure, while the investment powerhouses Goldman Sachs and JP Morgan have even started working on the full provisioning of cryptocurrency investments opportunities to their clients (Mason, 2021; Ponciano, 2021).

Moreover, many blockchain-based digital assets or *tokens* with, for instance, the purpose of low volatility (*stablecoins*) and access to services (*utility*) (Oliveira et al., 2018) are booming in what has become popular under the term *decentralized finance (DeFi)* (Zetzsche et al., 2020). In general, the opportunities related to blockchain-based financial markets and tokenization are now regarded as a key trend for the economy (Alt, 2020; Sunyaev et al., 2021). IS researchers have early also investigated the opportunities of adopting blockchain technology beyond the financial sector and expected substantial improvements, e.g., in terms of data immutability, interoperability, and traceability (Beck et al., 2018; Ferdous et al., 2019). Moreover, the opportunity to enforce rules between business parties on a blockchain can facilitate a new level of trust and, to some extent, make blockchains a substitute for intermediaries (Alt, 2020; Beck et al., 2017; Bons et al., 2020). Researchers and practitioners have explored blockchains in numerous publications and prototypes within, among others, supply chain management (Gonczol et al., 2020; Queiroz and Wamba, 2019) and the energy, health, mobility, and public sector (Andoni et al., 2019; Fridgen et al., 2019; Shi et al., 2020; Warkentin and Orgeron, 2020).

However, compared to the momentum of blockchain applications in cryptocurrencies and DeFi, adoption in industry and the public sector seems to move considerably slower. For instance, besides a few successful, productive solutions (Lacity and van Hoek, 2021), we have not yet observed the anticipated widespread disruption of digital supply chain management. Considering the large number of publications and businesses' significant efforts to develop blockchain-based solutions beyond the financial sector (International Data Corporation, 2021), the visibility of successful blockchain applications seems relatively limited. During the Covid-19 pandemic, we also saw many blockchain-related projects being placed on hold or quit, possibly owing to a lack of success and the shift in priorities toward other projects that promise short-term savings or that open new business opportunities. Insights from large consultancies support this observation. For instance,

Challenges for Blockchain Adoption	**Example references**
Alignment with business models and services	Heines et al. (2021), Janssen et al. (2020), Toufaily et al. (2021)
Integration into organizations' legacy systems	Alt (2020), Babich and Hilary (2020), Sedlmeir et al. (2022),
Heterogeneous levels of digitalization	Fridgen et al. (2018), Jensen et al. (2019)
Compliance with legal frameworks and institutional processes	Janssen et al. (2020), Lacity (2018)
Governing collaboration among stakeholders	Beck et al. (2018), Lacity and van Hoek (2021)
Closing communication gaps	Sedlmeir et al. (2020)
Scalability and performance	Kannengießer et al. (2020), Sedlmeir et al. (2022), Toufaily et al. (2021)
Correctness and updatebility of code	Kannengiesser et al. (2021), Köhler and Pizzol (2020)
Transparency of sensitive data	Kannengiesser et al. (2021), Pedersen et al. (2019), Toufaily et al. (2021),

Table 1: Organizational challenges of blockchain adoption as pointed out by extant research.

Deloitte recently found that the mortality rate of blockchain projects pursued by organizations is around 85 %, and even 92 % when taking into account all blockchain projects on GitHub (Deloitte, 2021). Further, large technology companies such as IBM and Microsoft have announced a reduction in their blockchain engagements (Allison, 2021). A high failure rate for large and complex IT projects is not surprising per se (Whitney and Daniels, 2013), and an even higher failure rate may be expected owing to a certain level of blockchain hype associated with financial speculation in the context of cryptocurrencies and DeFi. Nonetheless, the observation of unexpectedly slow developments regarding blockchain adoption beyond concepts and prototypes has already led to disillusionment and nascent research on why blockchain technology has to date failed to meet the high initial expectations in the context of supply chains (Sternberg et al., 2020). Given that particularly the connecting of today's fragmented information silos in supply chains was regarded as one of the very promising use cases for blockchains (Azzi et al., 2019; Queiroz et al., 2019; Roeck et al., 2019; Saberi et al., 2018), the lack of productive solutions there is particularly surprising.

Table 1 features a summary of challenges that organizations face in blockchain adoption. In this paper, we argue why we consider *excessive* transparency one of the key reasons for the observable lack of blockchain adoption. Building on previous work, we discuss why the use of a blockchain often conflicts with organizations' policies and regulations associated with sensitive business and customer information (Kannengiesser et al., 2021; Pedersen et al., 2019; Toufaily et al., 2021). The impracticality of deleting data ex-post from a practically immutable ledger further aggravates these issues (Rieger et al., 2019). Initial calls for research into the privacy implications of blockchains have pointed out that researchers should explicitly consider issues associated with the exposure of sensitive information (Rossi et al., 2019). In this context, Kannengießer et al. (2020), for instance, have already contributed to a more detailed understanding of the related trade-offs from a technical perspective. Yet, we found that transparency-related discussions are often restricted to personal information and the GDPR's *right to be forgotten* (Schellinger et al., 2022) or not considered a substantial challenge (e.g., Lacity and van Hoek, 2021). Some researchers even consider blockchain as a suitable technology to increase privacy (e.g., see the overview in Karger, 2020). During our involvement in more than 10 projects in the mobility, energy, and public sector in the last three years in which we designed, implemented, and evaluated blockchain-based solutions, we initially encountered similar perspectives among stakeholders, which also aligns with the findings by Platt et al. (2021). In these projects, the exposure of sensitive information often made scaling blockchain-based applications from initial proofs of concept to larger ecosystems very difficult, required substantial architectural changes, and caused increased complexity or restricted the originally intended scope.

To provide a shared understanding of the application areas of blockchain technology that we use to illustrate the consequences of excessive transparency, we first introduce some background on blockchain technology, derive common use case patterns, and list examples for the sensitive information involved. We then point out the fundamental transparency challenge that affects many of these patterns and the corresponding difficulties developers and decision-makers face in businesses and institutions when conceptualizing or scaling corporate blockchain applications. We also illustrate to which extent permissioned blockchains and some recent developments in the practical use of cryptographic tools may help mitigate the transparency challenge. We close by summarizing our main results and identifying avenues for future research.

G.2 Background

A blockchain is a specific distributed ledger type that builds on a peer-to-peer network where all data are replicated across multiple servers (*nodes*) in a fault-tolerant way (Butijn et al., 2020). Blockchains' physically distributed and organizationally decentralized yet logically synchronized data management is achieved through an append-only structure in which batches of transactions (*blocks*) are linearly connected through hash-pointers (*chain*) (Beck et al., 2017). Nodes decide which blocks to append and how to order the transactions within a block through a *consensus mechanism* (Wüst and Gervais, 2018). Provided a majority of the network in a specific metric such as hash rate (*proof of work*), the share of cryptocurrency (*proof of stake*), or the number or reputation of nodes (*voting-based* or *proof of authority* consensus) is honest, this guarantees the correct execution of transactions and the practical immutability of the ledger. Transactions can represent a simple payment or the execution of a program (*smart contract*) whose code is specified through a previous transaction (Butijn et al., 2020). The confidence that the execution of a transaction has the intended consequences and cannot retrospectively be altered without the need to rely on the availability and honesty of a specific entity is often referred to as digital trust (Nofer et al., 2017).

A common categorization distinguishes between *permissionless* blockchains, where any entity can participate in consensus, and *permissioned* blockchains, where only selected entities can take this role, for instance, within a consortium from industry or the public sector (Beck et al., 2018; Wüst and Gervais, 2018). Permissionless blockchains are *public*, i.e., any entity can download and read the corresponding state of the ledger. By contrast, permissioned blockchains are often – but not always – *private*, i.e., only authorized entities have read access (Rossi et al., 2019). As active participation in consensus typically involves receiving, reading, storing, and executing transactions and updating the local ledger accordingly, the nodes participating in consensus are a subset of the entities with reading access. It is also important to note that in this sense, many blockchains used in the public sector are private and permissioned, as they are run by and accessible to selected entities only (Rieger et al., 2019).

The enforcement of business logic through smart contracts technically prevents misconduct by individual participants and creates trust in the correct handling of processes (Bons et al., 2020). For instance, the Ethereum blockchain can even be considered a platform of platforms, specifically for financial applications (Buterin et al., 2014) but intended for more general purposes. Blockchain-based information systems for use in organizations

#	Pattern	Example use cases	References	Types of sensitive information
1.	Payment	Bitcoin, Central bank digital currencies	Nakamoto (2008), Dashkevich et al. (2020)	Individuals' and businesses' revenues, expenses, balances, turnover and business partners
2.	Tamper-proof documentation	Notarization, Cardossier	EC (2021), Zavolokina et al. (2020)	Content and validity status of documents, information that could be sold on a market
3.	Cross-organizational workflow management	Tradelens, MediLedger	Jensen et al. (2019), Mattke et al. (2019)	Frequency and type of processes, relationships between organizations involved
4.	Ubiquitous services	Oracles (Chainlink), DeFi (Uniswap)	Al-Breiki et al. (2020), Wang et al. (2019), Werner et al. (2021)	Risk exposure associated with financial investments
5.	Digital identities	Namecoin, German asylum case	Kalodner et al. (2015), Amend et al. (2021)	Individuals' names, addresses, health information, permissions and achievements
6.	Tokenization	Ticketing (GUTs), investments and fractional ownership	Regner et al. (2019), Sunyaev et al. (2021), Whitaker and Kräussl (2020)	Individuals' and organizations' investment decisions and voting behaviour
7.	Machine economy	Micropayments, economically autonomous robots	Jöhnk et al. (2021), Schweizer et al. (2020)	All of the above; machines are typically associated with organizations or individuals

Table 2: Blockchain application patterns and examples for the sensitive information involved.

can also be seen as an alternative to a trusted third party – for instance, if stakeholders cannot agree on a potential platform owner because they fear its corresponding market power. Blockchains and smart contracts also provide the opportunity to implement a variety of applications that involve multiple organizations on the same neutral platform with strong guarantees on the correctness and non-repudiability of transactions (Bons et al., 2020; Fridgen et al., 2019). Yet, it is unlikely that blockchains can provide a purely technical substitute for all services established trusted intermediaries provide today (Fridgen et al., 2021).

Beyond this commonality, blockchain applications are very heterogeneous and can be associated with many different use cases. While research has already provided different classifications, often with a fairly technical focus (e.g., see Xu et al., 2018), so far there has been no focus on the types of sensitive data involved. We hence present some *use case patterns* (payment, tamper-resistant documentation, cross-organizational workflow management, ubiquitous services, digital identities, tokenization, and machine economy) to illustrate what kind of sensitive information they can involve. We will repeatedly use these use case patterns, which we summarize in Table 2, to illustrate related transparency challenges and solution approaches in Sections G.3 and G.4.

1. *Payment*

   Likely the best-known application of blockchain technology is digital payments. In this context, the cryptocurrency Bitcoin is a popular and arguably the foundational example (Nakamoto, 2008). Many stakeholders also consider smart contract-enabled conditional payments to be an appealing application. Blockchain technology has also been tested to improve traditional payment systems' efficiency, for instance, by easing inter-bank settlement, or for digital currencies directly issued by the central bank (Dashkevich et al., 2020). These examples can involve sensitive information such as individuals' and businesses' revenues, expenses, balances, turnover, or metadata that reveals the frequency of interactions between businesses and individuals.

2. *Tamper-resistant documentation*

   Trust plays a key role in payment transactions and is facilitated through the practical immutability of information stored on blockchains. However, tamper-resistant data storage can enable applications beyond payments to prevent – or at least make evident – the ex-post manipulation of processed information. For instance, one of the four core use cases for the European Blockchain Services Infrastructure is notarization, seeking to provide a service for creating trusted digital audit trails that allow one to prove the integrity of diplomas or administrative documents (European Commission, 2021). Another application area for tamper-proof documentation is Cardossier, which allows one to collect and sell verifiable data about used cars, thus reducing information asymmetries in markets (Zavolokina et al., 2020) and increasing consumer trust (Bauer et al., 2019). Therefore, the recorded data can be personally identifiable or have business value.

3. *Cross-organizational workflow management*

   The availability of an infrastructure for tamper-resistant documentation and the timely distribution of information to many parties also enable the cross-organizational coordination of business processes. Smart contracts can enable event handling, facilitating process control, and, in the long term, the automation of selected process steps within cross-organizational business relationships (Fridgen et al., 2018; Sturm et al., 2019). The coordination of such processes requires the visibility of information such as the time, frequency, and utilization of services or processes, to third-party organizations to enable cross-organizational workflow management (Kannengiesser et al., 2021). One prominent example in the logistics sector is TradeLens, a blockchain-enabled platform that aims to improve the scheduling along the maritime logistics chain by communicating shipping events while tracking shipping containers and digitizing the related documentation (Jensen et al., 2019). Another example of a permissioned blockchain is MediLedger, which prevents the injection of fake medicals in pharmaceutical supply chains through improved information exchange between various stakeholders and preventing the *double-spending* of authentic medicals (Mattke et al., 2019).

4. *Ubiquitous services*

   Many services on blockchain-based platforms are available even without the need to interact with a business or another organization. These ubiquitous services are provided through smart contracts. Once published, smart contracts typically remain available without further maintenance by the original developer as long as the underlying blockchain continues to be operated; thus, they can offer *services without service providers*. One prominent example is automated market makers that facilitate decentralized exchanges through providing a pricing mechanism in a smart contract, for instance, Uniswap, or managing investment portfolios in DeFi (Grigo et al., 2020; Werner et al., 2021). Another popular kind of ubiquitous services are *oracles*, which provide information from the external world, such as stock prices, meteorological data, or flight delays, on-chain. Oracles are also implemented via smart contracts and often employ *truth discovery* methods that compare different inputs and involve combinations of incentives and penalties to make the provided data reliable (Al-Breiki et al., 2020).

5. *Digital identities*

   The provision of digital identities can be regarded as a particularly impactful application for ubiquitous services. In many applications, digital representations of physical entities are needed (Dietz and Pernul, 2020). Blockchains' transparency and tamper resistance have been used early on to link entities to public keys (Kalodner et al., 2015). On the other hand, blockchain technology has also popularized the concept of a *digital wallet* that organizations, users, and smart things can maintain to claim not only the ownership of cryptocurrencies but also of digital identities that verifiably attest their attributes and authorizations. Germany's Federal Office for Migration and Refugees is already active in this area and is investigating the possibility of creating a unique digital identity for refugees that is suitable for administrative purposes across organizational boundaries (Amend et al., 2021).

6. *Tokenization*

   Besides unique identities for persons, organizations, and machines, blockchains can also create digital representations of scarce physical and digital assets. However, in this context, the emphasis is not on allowing these objects to maintain their own identity but rather to make them tradable with a global pool of potential buyers. While fungible tokens, such as units of a cryptocurrency, are interchangeable, non-fungible tokens (NFTs) are digital representations of unique physical or digital objects, such as collectibles, artworks, or virtual gaming assets. The change of ownership relationships and attributes of such tokens are recorded on blockchains. NFTs can represent tickets (Regner et al., 2019), real estate, services, artwork, or other creative work. An illustrative example is GUTS, an event ticketing system that empowers visitors to exercise full control over their tickets, including reselling them, while giving the event organizer secondary market control in terms of prices. *Tokenization* also enables fractional ownership, thereby potentially increasing previously illiquid markets' liquidity (Whitaker and Kräussl, 2020) and allowing investors to vote on how the underlying asset should be managed.

7. *Machine economy*

   Ultimately, machines can maintain their own identity and exchange value through tokens. Micropayments can improve processes between various machine entities. Owing to rapid developments in artificial intelligence and the Internet of Things, it is likely only a matter of time before machines can interact autonomously with one another (Jöhnk et al., 2021). With the absence of centralized monitoring and decision-making, a blockchain can serve as a trust-based technology and infrastructure to enable the exchange of master data, dynamic data but also digital assets between such autonomous agents (Schweizer et al., 2020).

G.3 The transparency challenge

a) Problem statement

In public permissionless blockchains, every block, including all transactions to be operated, is generally disseminated to every node. Nodes then store and check each transaction and compute the corresponding updates to the *world state* – a running aggregate representation of all previously executed transactions that is maintained for efficiency reasons.[1] This inherent redundancy of data processing and storage in blockchains facilitates fault-tolerance through cross-checking and forms the backbone of blockchains' promise of providing digital trust. On the other hand, replication by a large number of nodes, some of which may not be trustworthy, is a double-edged sword: it inevitably leads to challenges associated with the exposure of sensitive information such as critical business data or personally identifiable user data (Platt et al., 2021; Zhang et al., 2019).

So far, transparency concerns seem to play only a minor role in cryptocurrencies and related financial applications of blockchain. As it is known that users' pseudonymous blockchain addresses can often easily be mapped to natural persons or organizations (Biryukov and Tikhomirov, 2019), essentially, today individual users or companies are deciding wittingly to reveal their transactions and, thus, their payments, investments, strategies, and risk exposure. Nonetheless, excessive transparency is currently a major challenge for DeFi from another perspective: block-producing nodes can not only decide which transactions to include in the next block but also in which order. Hence, they can make additional profit by observing the transaction proposals that have not yet been included in a block (the *mempool*) and selecting and ordering them in their favour or even *sandwiching* them between own transactions that are only conducted for this reason to make arbitrage (Daian et al., 2020). This is not only problematic from a regulatory perspective and typically forbidden in regulated markets (McCann, 2000), it can also lead to misaligned incentives in consensus that reduce the security of the underlying blockchain infrastructure.

In many applications, the disclosure of data to other blockchain nodes by default often conflicts with companies' data policies, customers' expectations, and antitrust and data protection regulations, and specifically with the GDPR's "right to be forgot-

[1] Replication is also typical of many other kinds of distributed ledgers with alternative data structures, like, for example, directed acyclic graphs, and many aspects of our discussion hence extend to these, too. However, for simplicity, we will stick to blockchains for the remainder of this paper.

ten" (Schellinger et al., 2022). While individuals can agree with the processing and sharing of their data, they can demand deletion at a later stage according to the GDPR. As organizations expected benefits from the sharing of verifiable personal information via digital identities to streamline processes, this dilemma has resulted, for instance, in the development of workarounds that allow one to remove data retroactively despite the presumed immutability of blockchains (e.g., Ateniese et al., 2017; Deuber et al., 2019). Nonetheless, enforcing the deletion of all copies that nodes may have made is technically impossible. Further, if it is necessary to undertake major efforts to delete supposedly confidential data on a blockchain, it may not have been a good idea to replicate them among multiple nodes in the first place. On the other hand, the GDPR also lists requirements such as *purpose limitation* and *privacy by default* (Haque et al., 2021; Schellinger et al., 2022) that makes already the initial replication of data by multiple organizations – many of which are unlikely involved in the associated process – questionable. Thus, although Bélanger and Crossler (2011) generally advises that one study information privacy issues at the "organization level," it seems justified to specifically consider the implications of using blockchain technology on data visibility.

Similar considerations apply for sensitive business information: Enterprises that wish to lever a blockchain for use case patterns such as cross-organizational workflow management to share data or to improve the coordination of fragmented, multi-lateral business processes hence need to think through the potential consequences of exposing business-critical data on a blockchain in detail. For instance, consider a cross-organizational workflow process. If information such as a part ID associated with this workflow is stored on a blockchain, at least all participants that run a node will have access to these data and often will be able to infer which entity was involved in manufacturing steps related to this part ID because transactions are digitally signed, and repetitive patterns can help with the de-pseudonymization of accounts. On the other hand, if data like part IDs are not stored on-chain, the process cannot be coordinated seamlessly through a smart contract owing to the lack of information that each of the parties would need for an end-to-end verification of provenance (Bader et al., 2021). This includes *qualitative* proofs of provenance that show that all the suppliers who contributed to a composite part were certified, which relates to organizations' digital identities. On the other hand – and arguably even more complicated – there are *quantitative* proofs of provenance, for instance, to demonstrate that a business only uses ethically sourced precious metals or green energy for a specific product. This topic is increasingly relevant in the context of regulation like the novel European supply chain law, which was, for instance, recently followed by the German

Supply Chain Act (German Federal Government, 2021), and the increasing demand for holistically tracking carbon emissions that a specific product has caused across its supply chain (Sundarakani et al., 2010). Research has already suggested to use blockchain technology to monitor resource usage in production and logistics (Manupati et al., 2019), and representing resources by tokens seems to be a viable approach to prevent double-usage. However, in both cases, stakeholders will see a lot of information about other entities and their actions in the supply chain who are not their direct business partners.

b) Encryption and hashing only helps in limited scenarios

Many blockchain projects have decided to mitigate privacy issues by putting the data on a blockchain only in encrypted or hashed form. By this method, consensus can be found on obfuscated data that can still be used to prove the integrity of the original data without the need to replicate it directly on the blockchain (Schellinger et al., 2022). Yet, it is also risky and inefficient to publish specifically encrypted data on a blockchain: While conventional software and databases can regularly update their encryption algorithms to keep up with new developments and threat scenarios and also periodically re-encrypt it with a new, more secure algorithm, the immutability of a blockchain's ledger implies that historic encrypted data is exposed to all nodes without such modifications. Consequently, blockchains may pose a tempting target for future decryption attacks with brute force (Xu et al., 2021) or quantum computers (Lindsay, 2020). Even hashed identity information on a blockchain can be problematic, specifically if referred to repeatedly (Finck, 2018; Marx et al., 2018).

Both encryption and hashing also make data largely useless as inputs for smart contracts since checking conditions or performing other computations typically conducted by smart contracts is generally not possible on obfuscated data.[2] To utilize the proclaimed benefits of smart contracts, the code itself, input, and output data need to be accessible to the other blockchain nodes (Kannengiesser et al., 2021). For instance, looking at the use case patterns of payment and cross-organizational workflows, the approach to handle business logic such as conditional payments or auctions using smart contracts implies that the data that underlies these operations (e.g., the variables on which conditional checks are performed, or ownership relationships) need to be available on-chain because otherwise,

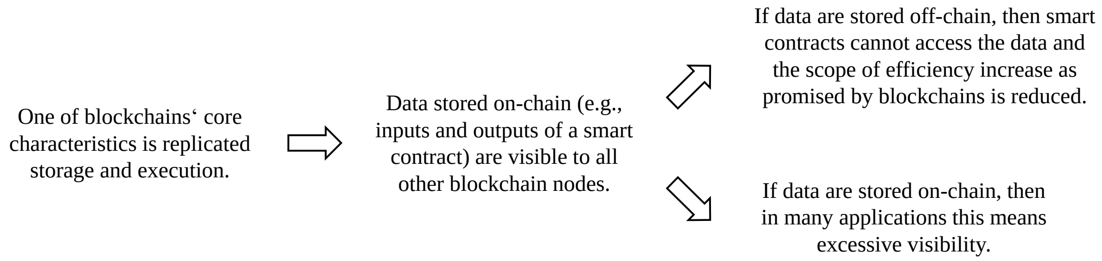
[2] We do not discuss homomorphic encryption here because at the moment, we consider it too specific (partial HE) or too computationally intensive (fully HE) to be practical on blockchains as of today beyond a few special cases.

the nodes cannot validate a new transaction by computing its impact on the world state and cannot update their local ledger accordingly. However, this data sharing with other nodes by default may not be in the interest of a party writing the code or holding the input data (Platt et al., 2021). Thus, while tamper-resistant documentation can be achieved without major privacy challenges and trade-offs, it is unclear how coordinating or automating processes that require the provision of multiple parties' inputs in smart contracts should be achieved without excessive transparency.

c) The fundamental tradeoff between restricted visibility and efficiency

This dilemma inhibits many use cases in which the information that is necessary to automate processes on a blockchain may not be revealed to other parties for corporate secret (*need to know*) or antitrust regulation reasons. It also makes businesses such as suppliers whose business model is based on information asymmetries reluctant to join a blockchain-based platform that would reveal their business relationships and processes to upstream- and downstream entities and competitors. This issue is particularly unfortunate since the collaboration between many potentially competing businesses on a neutral platform was thought to be one of the areas where blockchain technology has the highest economic potential. While reducing information asymmetries can be beneficial, revealing potentially sensitive business and customer information to competitors and other third parties is often so problematic that it inhibits uploading business-related data to a blockchain entirely.

Compared to other often-mentioned challenges of blockchain diffusion, there is also an interesting abstract argument why the transparency challenge seems special: issues such as integration with legacy systems, governance, or performance can be solved *incrementally* by gradually increasing the scope of processes and the number of participants in the system, by optimizing protocols and code, or by improving compute power and bandwidth over time (Sedlmeir et al., 2021a). In contrast, information that is shared on a blockchain has another quality: either a piece of information is written to the blockchain and therefore available to the other nodes, or it is not. Beyond a few special cases of statistical information disclosure techniques such as differential privacy in big data (Bugliesi et al., 2006), it seems an open question how data can be made *incrementally less sensitive* while at the same time being useful as inputs of a smart contract that, for instance, conducts a conditional check.



One of blockchains‘ core characteristics is replicated storage and execution.

Data stored on-chain (e.g., inputs and outputs of a smart contract) are visible to all other blockchain nodes.

If data are stored off-chain, then smart contracts cannot access the data and the scope of efficiency increase as promised by blockchains is reduced.

If data are stored on-chain, then in many applications this means excessive visibility.

Figure 1: The core argument why there is a transparency challenge for blockchains.

Thus, we observe a seemingly fundamental trade-off between efficiency gains and excessive data visibility issues (see Figure 1). A focus on the operation of business logic and the automation of processes via smart contracts requires storing related input and output data for the smart contract on-chain, which causes issues with the compliant handling of sensitive data. On the other hand, reducing the amount of information that is available on-chain means that there is less information to use in smart contracts and thus reduced utility from the blockchain. This main privacy challenge can be regarded as an economically oriented version of the trade-off *Turing-complete smart contracts versus data confidentiality* as presented in Kannengießer et al. (2020), and has been acknowledged – albeit often with less emphasis – by many research articles on blockchain technology (e.g., Toufaily et al., 2021).

G.4 Solution approaches

In this Section, we illustrate three approaches – permissioned blockchains, self-sovereign identities for individuals and organizations, and verifiable computation focusing on zero-knowledge proofs – that can help avoid excessive information exposure on blockchains.

G.4.1 Permissioned blockchains

One natural reaction of businesses to challenges relating to public permissionless blockchains, which besides excessive data visibility include low throughput, relatively high confirmation latencies, and high and often volatile transaction costs (Sedlmeir et al., 2022), is moving to private permissioned blockchains that restrict read access and participation in consensus and therefore provide better control of information exposure. This approach has, therefore, often been advised as a satisfactory solution to privacy issues (e.g. see Lacity and van Hoek, 2021). However, permissioned blockchains can only partially mitigate the fundamental transparency challenge since exposing sensitive information only to a few other stakeholders can still be an inhibiting problem. For instance, TradeLens even levers multiple blockchains (*channels*) to separate the large and competing shipment carriers from one another and to avoid that a large carrier can count the events associated with another carrier and learn about how its business is going. Nonetheless, within one channel, there are still many potentially competing stakeholders such as ports and logistics service providers, and information that is sensitive from the perspective of clients – such as the Bill of Lading – needs to be stored off-chain (Jensen et al., 2019). Thus, for instance, the information registered in the Bill of Lading cannot be used for managing escrows or market activity on the blockchain-based solution.

To further mitigate the negative consequences of excessive transparency, popular permissioned blockchains such as Hyperledger Fabric and Quorum support *private transactions* (Consensys/GoQuorum, 2021; Guggenberger et al., 2022). In these private transactions, hashed or encrypted data are distributed to all nodes, and only selected nodes specified on the smart contract or transaction level perform the execution based on the original data that they can request through a peer-to-peer messaging layer or read from the blockchain and decrypt. Similar approaches can be made on permissionless blockchains by specifying that for valid updates to a smart contract state, only the signatures of selected parties on the updated state or a commitment onto it are required. Involving all parties affected by a specific transaction reduces information exposure without a trade-off in trust. However, the restricted access to information on-chain again implies that a smart contract can only offer considerably less functionality or that another communication layer needs to be added to distribute the underlying data between the involved entities. For instance, if a blockchain is meant to be used for the traceability of components in the automotive supply chain such that all cars containing one part from a problematic delivery of a Tier n supplier can be determined, this means that all information about the fabrication of sub-components and their provenance needs to be visible at least upstream. Since information asymmetries in supply chains are essential for most suppliers' business models, it is not surprising that blockchains have a tough time in such use cases where the splitting and merging of components along the supply chain are more complex than tracking the route of a container or a charge of largely unprocessed groceries or products, as in IBM's seemingly successful Food Trust (Kamath, 2018). Essentially, the core transparency challenge hence remains also in the private permissioned setting: the more utility smart contracts are supposed to offer, the more daunting the challenges related to the disclosure of sensitive information.

Besides, switching to a permissioned blockchain also comes at additional disadvantages, as setting up and maintaining nodes for a domain-specific permissioned ledger requires skilled employees, much coordination effort, and a sophisticated governance mechanism that enterprises need to invest in. Moreover, different permissioned blockchains are difficult to connect, so using many fragmented permissioned blockchains can substantially decrease the network effects that proponents of blockchain technology have expected (Brody, 2019). Indeed, the results of a recent study by Toufaily et al. (2021) indicate that organizations tend to switch from permissioned to permissionless blockchains. Consequently, permissioned blockchains are not a general solution to the transparency challenge.

G.4.2 Digital identities

a) Self-sovereign identities for individuals As previously discussed, the replicated storage of personal information does not comply with privacy regulation like the GDPR and hence makes storing digital identity information directly on a blockchain practically impossible for organizations. Fortunately, the immutability of identity-related information as one of the core value propositions expected from blockchains can be provided in many cases by third parties' digital signatures (Sedlmeir et al., 2021b). For instance, federal printers that issue digital ID cards or universities that provide digital diplomas are typically trusted in their specific, limited domain. Immutability alone is also often not sufficient for identity documents, because also the authenticity of the information at the time of writing is relevant; for instance, that a Covid-19 vaccination credential was issued by a certified doctor (Rieger et al., 2021). On this basis, many projects that focus on privacy and user-oriented identity management or the bilateral exchange of information don't use a blockchain for the storage of identity-related information or hashes thereof. Rather, they only involve a distributed ledger as a substitute for specific, ecosystem-related services that have so far been provided by certificate authorities and that involve information that is meant to be public (Schlatt et al., 2022). Early examples of this approach are Canada's Verifiable Organizations Network and Germany's IDunion consortium. This decentralized or self-sovereign identity (SSI) paradigm was largely motivated by the digital wallets that became popular through blockchains and is also often affiliated with blockchains (Čučko and Turkanović, 2021; Soltani et al., 2021). In this sense, despite the high sensitivity of involved personal data, digital identities may be one of the few blockchain application patterns with no significant privacy challenges because the main data exchange happens in bilateral communication in the form of digital certificates, and the blockchain only provides a tamper-resistant ledger for public data such as issuers' signing keys and implementing technical governance mechanisms.

The availability of digital and verifiable data for users and institutions is not only a promising application *of* blockchain that does not exhibit privacy issues to the extent of other patterns, but also allows one to transfer information and corresponding existing *real-world* trust frameworks *to* blockchains in a verifiable way. Many business-related use cases will require the feed-in of verifiable off-chain data, such as a proof of legal age or of accomplished tax payments, in the future. Another application area is the verifiability of sensor data utilizing a certificate that confirms the sensor's provenance and proper calibration. Here, digital identity management may offer an alternative approach to oracles (Caldarelli, 2020) and replace truth discovery mechanisms through the verifiability of cryptographic proofs of provenance. Moreover, this also provides the opportunity to selectively disclose information from a larger, verifiable dataset: The privacy capabilities used in many SSI implementations for the selective disclosure of attributes can even provide the data minimization or anonymization required for natural persons to directly interact with smart contracts while complying with regulation (Platt et al., 2021). Thus, approaches to decentralized identity management where blockchain technology only plays a moderate role can likely become the key building block in many applications that were thought to be a core blockchain case but may also help to connect blockchains with real-world identity and trust frameworks, extending their capabilities.

b) Self-sovereign identities for organizations The availability of digital identities for organizations also enables efficient cross-organizational identification and, thus, authenticated bilateral data exchange. This may improve the exchange of both master data and dynamic data between enterprises (Hyperledger-Labs, n.d.). Based on such solutions, organizations can manage other organizations' permissions in a fine-grained way, facilitating an access management for bilateral (non-blockchain based) operational data exchange that satisfies data sovereignty and interoperability requirements. For this reason, digital identities for organizations will likely play an important role in the European cloud initiative GAIA-X.

The bilateral exchange of authentic information between organizations should be considered as a prerequisite for blockchains rather than a consequence: it allows stakeholders to communicate sensitive data that are not suitable to store on a blockchain but that may be necessary to make sense of otherwise obfuscated, blockchain-based transactions and events (e.g., in the form of hashes). Once there is a solid foundation for bilateral communication, data related to relevant processes or the need to interact with other stakeholders can *selectively* be taken to higher transparency so as to add further utility. An all-or-nothing approach can hardly be regarded as suitable in a system in which the degree of transparency needs to be well-balanced. Moreover, the anonymization and selective disclosure features of SSI can also help organizations coordinate workflows on-chain without leaving a trace of sensitive information.

The situation that current SSI initiatives lever cryptographic methods such as public key cryptography that is also incorporated in blockchains and that require sophisticated cryptographic key management, and that most of them even build on a blockchain instead of certificate authorities, may also allow enterprises to become familiar with technical and organizational best practices for wallet usability and the development and governance of decentralized applications in production. Further, if designed as discussed, the use cases of digital identities on the one side and payment and tokenization on the other side may be complementary: Blockchain technology's supposed initial core value proposition was the transfer of value in the form of cryptocurrencies or tokens across multiple stakeholders without an intermediary. This transfer of value cannot be solved by the digital certificates employed in SSI, since they can be copied and used repeatedly. On the other hand, digital certificates allow stakeholders to exchange verifiable data bilaterally and, thus, avoid the storage of sensitive information on a blockchain. Yet, while SSI can provide an additional, standardized information exchange layer without intrinsic transparency issues and allows persons and entities to selectively and verifiably reveal authorizations and attributes as attested by third parties also on-chain, many limitations do not make it a general solution for the transparency challenge. For instance, SSI cannot help in many scenarios where a third-party attestation is not available or – as common in blockchain applications – not trusted by all relevant stakeholders.

G.4.3 Verifiable computation

a) Validation is possible without full knowledge In many use cases, blockchain nodes only need to know selective information about what is being processed in payments or smart contract operations to verify a transaction's validity. A simple example of a cross-organizational workflow management case is a logistics supply chain in which transactions should be visible to only a small subset of nodes or clients. This can be achieved, for instance, through attribute-based encryption that offers a convenient way to allow decryption only to a specific subset of participants on the blockchain, based on their digital identities (Bader et al., 2021). In permissioned blockchains, the previously discussed private transactions provide similar features. However, if a transaction changes a variable that may affect other parties, pure visibility restriction through encryption-based access control becomes less useful, and more complex privacy-enhancing technologies need to be applied. For instance, in a simple payment, if entity B wants to receive a payment from entity A, entity B needs to be able to verify that it received the intended amount, while all other stakeholders indirectly affected by this transfer (i.e., owners of units of the same kind of tokens) only need to be sure that entity A's balance is high enough to cover the transaction and that the total supply of token units is unchanged, since otherwise, the value of their own assets may decrease as a result. The transaction amount and A's and B's identities are irrelevant to the other stakeholders (excluding the regulator in this simple example).

Similar patterns are present in industry, where stakeholders or regulators want to be convinced that business partners comply with specific rules, while many other details are not relevant. A thriving cross-organizational workflow example from supply chain management is MediLedger, where pharmaceutical businesses (and ultimately, the regulator) require a proof that a delivery of medicals is authentic. If the sender can convince all blockchain nodes that this is the case, no further information is needed (Mattke et al., 2019). For proving the invariance of a global variable (e.g., the number of authentic medicals) under a transaction, it is sufficient to prove local invariance in a transaction that only changes local states. Consequently, a company that records all the transactions it was involved in could demonstrate to an auditor that more units of a specific good were not sold than previously received at any time. Yet, as there is typically no auditor that all participants on the blockchain trust, SSI is not a viable solution, and purely cryptographic technologies are often used in this context.

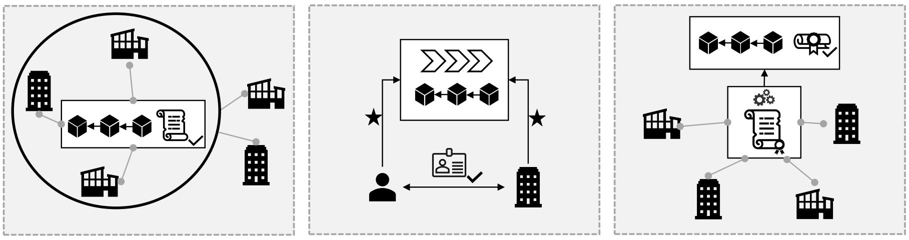
b) Zero-knowledge proofs One approach that has matured significantly over the last years are zero-knowledge proofs (ZKPs). ZKPs allow a *prover* to convince a *verifier* of the knowledge of data with specific properties (Goldwasser et al., 1989). One example could be that the prover proves to the verifier that he or she knows the solution to a Sudoku puzzle, without revealing any information that would make it easier for the verifier to solve the Sudoku puzzle him-/herself. A frequent type of proof that is relevant in the context of blockchains is a proof of knowledge of a pre-image of a hash (where the hash is public but the pre-image remains private), and a proof of knowledge of a digital signature that authorizes a transaction. More generally, ZKPs can be used to prove that some public data – which could itself be a hash – is the correct result of the execution of an algorithm on private data, without revealing any additional information (Ben-Sasson et al., 2014). ZKPs hence allow to replace the replicated execution of a transaction to ensure its integrity by the replicated execution of a proof verification algorithm that attests to the correctness of the result that was computed only by one entity. ZKPs can thus decouple the verifiability of data from their on-chain visibility (Platt et al., 2021). In the cryptocurrency Zcash, fully private (*shielded*) transactions are implemented with ZKPs (Ben-Sasson et al., 2014); and since ZKPs have also been used in many other blockchain-related projects to address data visibility challenges. For example, MediLedger took large parts of the Zcash implementation and adapted it to prove the authenticity of pharmaceuticals (Mattke et al., 2019). Thus, ZKPs can mitigate issues related to the confidentiality versus integrity trade-off discussed by (Kannengießer et al., 2020) because they enable the replicated verification of transactions and, thus, trust in their integrity despite not disclosing sensitive information. Generally, it may not be a coincidence that the early adoption of new cryptographic technologies that were previously successfully tested in a cryptocurrency may be adopted by businesses without requiring exceptionally high R&D expenditures.

c) Further verifiable computation technologies However, caution is required: First, the practical adoption of ZKPs is still in its infancy and has limitations. To date, levering ZKP causes additional complexity and requires experts from cryptography to translate business logic into corresponding code. While the proof verification conducted by every node is typically *succinct*, i.e., it requires very little computational resources, the prover still needs to provide expensive hardware (Bootle et al., 2020). Second, ZKPs' scope is naturally limited because the prover locally needs all the information to perform the original computation and to derive the associated proof. Thus, ZKPs cannot be used generically for privacy in smart contracts if their execution is supposed to compute on or modify private data from multiple entities, so other techniques are needed (Buterin, 2014). One approach is to use trusted execution environments (TEEs) like Intel's Software Guard Extensions (SGX), which ensures transactions can only be encrypted within a secure domain within the CPU and generates attestations for the computation's correctness. This approach is already quite flexible and offers very good performance. However, in the past, researchers have frequently found vulnerabilities of TEEs; and there is a single point of failure (the manufacturer of the TEE), which can be particularly problematic for blockchains not only in terms of trust but also considering lock-in effects. For example, several projects that aim to establish privacy in blockchains build on SGX (Bao et al., 2020), but recently, Intel announced that they would not integrate SGX in their new generation of CPUs (Pezzone, 2022). A popular trustless cryptographic alternative is multi-party computation (MPC) which allows the joint evaluation of a function of many variables, where each party only knows their private variables and learns the result. MPC has also been intensively researched but to date still seems challenging from a complexity and performance perspective to adopt in general settings (Šimunić et al., 2021), specifically if they need to be complemented, for instance, by ZKPs to prove the result's correctness on-chain. Nonetheless, there have been some promising explorations in selected blockchain applications already.

Thus, among the privacy-enhancing cryptographic technologies at hand, verifiable computation with ZKPs is often regarded as the currently most mature technology to offer solutions to blockchains' privacy challenges. The Ethereum ecosystem has been particularly innovative, and related projects should be closely observed by enterprises that wish to be at the forefront of integrating innovative solutions. As the research progresses, in the long run, all the aforementioned privacy-enhancing technologies may contribute (and be required) to solve the trade-off between privacy and efficiency in smart contracts.

G.4.4 Summary

In sum, we found three main approaches to how organizational blockchain solutions can address the transparency challenge, which we represent in Figure 2. In our view, all three alternatives are valuable in practice. While the first and second options seem quite easy to implement, they also have a relatively restricted scope. On the other hand, the third approach is still very complex to implement today, and there is not yet a generic solution that allows organizations to integrate verifiable computation as easily as other software components. From a more abstract perspective, we learn that – while consensus provides the backbone for stakeholders' trust in blockchains – the replication of the underlying sensitive information on all nodes is often more related to availability guarantees. Permissioned blockchains and, within them, specifically private transactions, can customize the entities that need to agree for consensus on the validity and implications of a transaction, and verifiable computation can allow for a separation between consensus on the correctness of the transaction and the underlying transaction data.



Limit visibility (and scope) by restricting participation on the blockchain to selected entities.

Exchange sensitive information off-chain with digital wallets and coordinate workflows on-chain.

Process data off-chain with privacy-enhancing technologies and provide proofs of computational integrity on-chain.

Figure 2: Main approaches to address the transparency challenge.

G.5 Conclusion

Initially, blockchain technology was regarded as a promising and disruptive solution beyond the financial sector, aiming at facilitating the digitalization in business networks where multiple potentially competing stakeholders need to operate on a joint digital infrastructure and streamline workflows (e.g., Alt (2020) and Frizzo-Barker et al. (2020)). While public blockchains in cryptocurrencies and the rich ecosystem of smart contract-based solutions in DeFi have already been remarkably successful, large-scale blockchain applications in industry and the public sector are still rare. We consider the privacy challenge a considerable reason for this. Blockchains' inherent degree of transparency often conflicts with corporate confidentiality policies and data protection regulation. Mitigating these privacy issues by moving data *off-the chain* comes with reduced functionality and increased complexity since smart contracts can generally only operate on data that are available to all parties affected by their implications. Cryptographic solutions that address those main challenges are not one-size-fits-all and are often not yet practical or come with significantly increased complexity. This trade-off can be difficult to detect in an initially successful, often internal proof-of-concept that has disregarded privacy issues but becomes painfully apparent when scaling the use case to more business partners.

Consequently, the use of smart contracts – while appealing from a functional perspective – must be carefully considered owing to the trade-off between increased efficiency on the one hand and confidentiality issues on the other. Opportunities and risks associated with moving from a permissionless to a permissioned blockchain must also be pondered since permissioned blockchains can only partially address privacy challenges while at the same time carrying disadvantages in terms of additional efforts and a lack of interoperability with other blockchain-based projects. The need for increased global transparency may be the exception rather than the default for organizations, being desirable only where it complies with regulation or if its value outweighs the negative implications of revealing potentially competition-relevant information. Thus, we emphasize the need for a base layer for trustworthy and verifiable information exchange. Decentralized digital identities can help with this in two crucial ways: First, they can facilitate users' or smart devices' direct interaction with a smart contract through selective disclosure and make real-world trust frameworks available for the verification on blockchain solutions, which also provides verifiable data for a blockchain to address the *Oracle problem*. Second, building on standardized, cross-organizational identity management for businesses and institutions allows one to implement fine-grained yet efficient authentication and authorization policies and, therefore, to move the trustworthy exchange of sensitive data to another layer. Blockchains can become a beneficial tool in particular cases where bilateral data exchange needs to be supplemented by multi-stakeholder coordination, transparency, or auditability. Thus, SSI can play a central role in enabling blockchain adoption and its diffusion into practice. Ultimately, privacy-enhancing and verifiable computation technologies such as ZKPs that allow one to selectively disclose properties of transactions or processes while keeping data private will be a building block of many blockchain applications, and we recommend closely following the progress made in DeFi in these areas and to adopt mature approaches and implementation frameworks in organizations.

The present discourse reflects the multidisciplinarity that characterizes research into blockchain adoption in practice. There are multiple challenges and opportunities, and studying them provides many avenues for future IS research. Scholars and practitioners in the field need to be aware of developments in privacy-enhancing technologies in cryptography and assess new solutions' legal foundations and their compliance with antitrust and data protection regulations. The GDPR was often criticized as an inhibitor to innovation by the blockchain community. Yet, the case of identity management may suggest that strict privacy regulation can even contribute to finding a more appropriate technical role for blockchain in applications than initially foreseen. Nonetheless, the business perspective will ultimately decide which projects potential savings and new business opportunities justify investments in R&D and complex implementations. Deciding where to use centralized and decentralized components and how to complement them with privacy-enhancing technologies hence seems considerably more complex than what the early blockchain decision trees (e.g., Pedersen et al., 2019; Wüst and Gervais, 2018) have suggested; and designing guidelines is a promising avenue for IS researchers. In our view, blockchain research that considers technical, legal, and economic aspects is needed now more than ever, and there are rich opportunities for future work on blockchain diffusion.

Acknowledgment

We gratefully acknowledge the Luxembourg National Research Fund (FNR) and PayPal for their support of the PEARL Project *P17/IS/13342933/PayPal-FNR/Chair in DFS/Gilbert Fridgen* and the Bavarian Ministry of Economic Affairs, Regional Development and Energy for their funding of the project "Fraunhofer Blockchain Center (20-3066-2-6-14)" that made this paper possible. We also thank the editor and the anonymous reviewers for their highly valuable and constructive feedback.

References

Allison, I. (2021). *IBM blockchain is a shell of its former self after revenue misses, job cuts: Sources*. URL: https://www.coindesk.com/ibm-blockchain-revenue-misses-job-cuts-sources.

Alt, R. (2020). "Electronic markets on blockchain markets". In: *Electronic Markets* 30 (2), pp. 181–188. DOI: 10.1007/s12525-020-00428-1.

Amend, J., G. Fridgen, A. Rieger, T. Roth, and A. Stohr (2021). "The evolution of an architectural paradigm – using blockchain to build a cross-organizational enterprise service bus". In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 4301–4310. DOI: 10.24251/hicss.2021.522.

Andoni, M., V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum, and A. Peacock (2019). "Blockchain technology in the energy sector: A systematic review of challenges and opportunities". In: *Renewable and Sustainable Energy Reviews* 100, pp. 143–174. DOI: 10.1016/j.rser.2018.10.014.

Ateniese, G., B. Magri, D. Venturi, and E. Andrade (2017). "Redactable blockchain – or – rewriting history in Bitcoin and friends". In: *European Symposium on Security and Privacy*. IEEE, pp. 111–126. DOI: 10.1109/eurosp.2017.37.

Azzi, R., R. K. Chamoun, and M. Sokhn (2019). "The power of a blockchain-based supply chain". In: *Computers & Industrial Engineering* 135, pp. 582–592. DOI: 10.1016/j.cie.2019.06.042.

Babich, V. and G. Hilary (2020). "OM forum – Distributed ledgers and operations: What operations management researchers should know about blockchain technology". In: *Manufacturing & Service Operations Management* 22 (2), pp. 223–240. DOI: 10.1287/msom.2018.0752.

Bader, L., J. Pennekamp, R. Matzutt, D. Hedderich, M. Kowalski, V. Lücken, and K. Wehrle (2021). "Blockchain-based privacy preservation for supply chains supporting lightweight multi-hop information accountability". In: *Information Processing & Management* 58 (3), p. 102529. DOI: 10.1016/j.ipm.2021.102529.

Bao, Z., Q. Wang, W. Shi, L. Wang, H. Lei, and B. Chen (2020). "When blockchain meets SGX: An overview, challenges, and open issues". In: *IEEE Access* 8, pp. 170404–170420. DOI: 10.1109/access.2020.3024254.

Bauer, I., L. Zavolokina, and G. Schwabe (2019). "Is there a market for trusted car data?" In: *Electronic Markets* 30 (2), pp. 211–225. DOI: 10.1007/s12525-019-00368-5.

Beck, R., M. Avital, M. Rossi, and J. B. Thatcher (2017). "Blockchain technology in business and information systems research". In: *Business & Information Systems Engineering* 59 (6), pp. 381–384. DOI: 10.1007/s12599-017-0505-1.

Beck, R., C. Müller-Bloch, and J. L. King (2018). "Governance in the blockchain economy: A framework and research agenda". In: *Journal of the Association for Information Systems* 19 (10), pp. 1020–1034. DOI: 10.17705/1jais.00518.

Bélanger, F. and R. E. Crossler (2011). "Privacy in the digital age: A review of information privacy research in information ystems". In: *MIS Quarterly* 35 (4), pp. 1017–1041. DOI: 10.2307/41409971.

Ben-Sasson, E., A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza (2014). "Zerocash: Decentralized anonymous payments from Bitcoin". In: *Symposium on Security and Privacy*. IEEE, pp. 459–474. DOI: 10.1109/SP.2014.36.

Biryukov, A. and S. Tikhomirov (2019). "Deanonymization and linkability of cryptocurrency transactions based on network analysis". In: *European Symposium on Security and Privacy*. IEEE, pp. 172–184. DOI: 10.1109/eurosp.2019.00022.

Bons, R. W. H., J. Versendaal, L. Zavolokina, and W. L. Shi (2020). "Potential and limits of blockchain technology for networked businesses". In: *Electronic Markets* 30 (2), pp. 189–194. DOI: 10.1007/s12525-020-00421-8.

Bootle, J., A. Chiesa, and S. Liu (2020). *Zero-knowledge succinct arguments with a linear-time prover*. URL: https://eprint.iacr.org/2020/1527.pdf.

Al-Breiki, H., M. H. U. Rehman, K. Salah, and D. Svetinovic (2020). "Trustworthy blockchain oracles: Review, comparison, and open research challenges". In: *IEEE Access* 8, pp. 85675–85685. DOI: 10.1109/access.2020.2992698.

Brody, P. (2019). *How public blockchains are making private blockchains obsolete*. URL: https://www.ey.com/en_gl/innovation/how-public-blockchains-are-making-private-blockchains-obsolete.

Bugliesi, M., B. Preneel, V. Sassone, and I. Wegener (2006). "Differential privacy". In: *Automata, Languages and Programming*. Springer. DOI: 10.1007/11787006_1.

Buterin, V. et al. (2014). *A next-generation smart contract and decentralized application platform*. URL: https://github.com/ethereum/wiki/wiki/White-Paper.

Buterin, V. (2014). *Secret sharing DAOs: The other crypto 2.0*. URL: https://blog.ethereum.org/2014/12/26/secret-sharing-daos-crypto-2-0/.

Butijn, B.-J., D. A. Tamburri, and W.-J. van den Heuvel (2020). "Blockchains: A systematic multivocal literature review". In: *ACM Computing Surveys* 53 (3). DOI: 10.1145/3369052.

Caldarelli, G. (2020). "Real-world blockchain applications under the lens of the oracle problem. A systematic literature review". In: *International Conference on Technology Management, Operations and Decisions*. IEEE. DOI: 10.1109/ictmod49425.2020.9380598.

Consensys/GoQuorum (2021). *Private transaction lifecycle*. URL: https://docs.goquorum.consensys.net/en/stable/Concepts/Privacy/PrivateTransactionLifecycle/.

Čučko, Š. and M. Turkanović (2021). "Decentralized and self-sovereign identity: Systematic mapping study". In: *IEEE Access* 9, pp. 139009–139027. DOI: 10.1109/access.2021.3117588.

Daian, P., S. Goldfeder, T. Kell, Y. Li, X. Zhao, I. Bentov, L. Breidenbach, and A. Juels (2020). "Flash Boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability". In: *Symposium on Security and Privacy*. IEEE, pp. 910–927. DOI: 10.1109/sp40000.2020.00040.

Dashkevich, N., S. Counsell, and G. Destefanis (2020). "Blockchain application for central banks: A systematic mapping study". In: *IEEE Access* 8, pp. 139918–139952. DOI: 10.1109/access.2020.3012295.

Deloitte (2021). *Evolution of blockchain technology*. URL: https://www2.deloitte.com/us/en/insights/industry/financial-services/evolution-of-blockchain-github-platform.html.

Deuber, D., B. Magri, and S. A. K. Thyagarajan (2019). "Redactable blockchain in the permissionless setting". In: *Symposium on Security and Privacy*. IEEE, pp. 124–138. DOI: 10.1109/sp.2019.00039.

Dietz, M. and G. Pernul (2020). "Digital twin: Empowering enterprises towards a system-of-systems approach". In: *Business & Information Systems Engineering* 62 (2), pp. 179–184. DOI: 10.1007/s12599-019-00624-0.

European Commission (2021). *European blockchain services infrastructure*. URL: https://digital-strategy.ec.europa.eu/en/policies/european-blockchain-services-infrastructure.

Ferdous, M. S., F. Chowdhury, and M. O. Alassafi (2019). “In search of self-sovereign identity leveraging blockchain technology”. In: *IEEE Access* 7, pp. 103059–103079. DOI: 10.1109/access.2019.2931173.

Finck, M. (2018). “Blockchains and data protection in the European Union”. In: *European Data Protection Law Review* 4 (1), pp. 17–35. DOI: 10.21552/edpl/2018/1/6.

Fridgen, G., N. Guggenberger, T. Hoeren, W. Prinz, N. Urbach, J. Baur, H. Brockmeyer, W. Gräther, E. Rabovskaja, V. Schlatt, A. Schweizer, J. Sedlmeir, and L. Wederhake (2019). *Opportunities and challenges of DLT (blockchain) in mobility and logistics.* URL: https://eref.uni-bayreuth.de/44302/.

Fridgen, G., S. Radszuwill, A. Schweizer, and N. Urbach (2021). “Blockchain won’t kill the banks: Why disintermediation doesn’t work in international trade finance”. In: *Communications of the Association for Information Systems* 49, pp. 603–623. DOI: 10.17705/1cais.04932.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018). “Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation”. In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Frizzo-Barker, J., P. A. Chow-White, P. R. Adams, J. Mentanko, D. Ha, and S. Green (2020). “Blockchain as a disruptive technology for business: A systematic review”. In: *International Journal of Information Management* 51, p. 102029. DOI: 10.1016/j.ijinfomgt.2019.10.014.

German Federal Government (2021). *Supply Chain Act (Lieferkettengesetz).* URL: https://www.bundesregierung.de/breg-en/federal-government/supply-chain-act-1872076.

Goldwasser, S., S. Micali, and C. Rackoff (1989). “The knowledge complexity of interactive proof systems”. In: *SIAM Journal on Computing* 18 (1), pp. 186–208. DOI: 10.1137/0218012.

Gonczol, P., P. Katsikouli, L. Herskind, and N. Dragoni (2020). “Blockchain implementations and use cases for supply chains – a survey”. In: *IEEE Access* 8, pp. 11856–11871. DOI: 10.1109/access.2020.2964880.

Grigo, J., P. Hansen, A. Patz, and V. von Wachter (2020). *Decentralized finance (DeFi) – A new Fintech revolution? The blockchain trend explained.* URL: https://www.bitkom.org/sites/default/files/2020-07/200729_whitepaper_decentralized-finance.pdf.

Guggenberger, T., J. Sedlmeir, G. Fridgen, and A. Luckow (2022). “An in-depth investigation of the performance characteristics of Hyperledger Fabric”. In: *Computers and Industrial Engineering* 173, p. 108716. DOI: 10.1016/j.cie.2022.108716.

Haque, A. B., A. K. M. N. Islam, S. Hyrynsalmi, B. Naqvi, and K. Smolander (2021). "GDPR compliant blockchains – A systematic literature review". In: *IEEE Access* 9, pp. 50593–50606. DOI: 10.1109/access.2021.3069877.

Heines, R., N. Kannengießer, B. Sturm, R. Jung, and A. Sunyaev (2021). "Need for change: Business functions affected by the use of decentralized information systems". In: *Proceedings of the 42nd International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2021/fintech/fintech/16/.

Hyperledger-Labs (n.d.). *Business partner agent repository*. URL: https://github.com/hyperledger-labs/business-partner-agent.

International Data Corporation (2021). *Worldwide blockchain spending guide*. URL: https://www.idc.com/tracker/showproductinfo.jsp?containerId=IDC_P37345.

Janssen, M., V. Weerakkody, E. Ismagilova, U. Sivarajah, and Z. Irani (2020). "A framework for analysing blockchain technology adoption: Integrating institutional, market and technical factors". In: *International Journal of Information Management* 50, pp. 302–309. DOI: 10.1016/j.ijinfomgt.2019.08.012.

Jensen, T., J. Hedman, and S. Henningsson (2019). "How TradeLens delivers business value with blockchain technology". In: *MIS Quarterly Executive* 18 (4), pp. 221–243. DOI: 10.17705/2msqe.00018.

Jöhnk, J., T. Albrecht, L. Arnold, T. Guggenberger, L. Lämmermann, A. Schweizer, and N. Urbach (2021). "The rise of the machines: Conceptualizing the machine economy". In: *Proceedings of the 25th Pacific Asia Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/pacis2021/54/.

Kalodner, H. A., M. Carlsten, P. Ellenbogen, J. Bonneau, and A. Narayanan (2015). "An empirical study of Namecoin and lessons for decentralized namespace design". In: *14th Annual Workshop on the Economics of Information Security*. URL: http://www.econinfosec.org/archive/weis2015/papers/WEIS_2015_kalodner.pdf.

Kamath, R. (2018). "Food traceability on blockchain: Walmart's pork and mango pilots with IBM". In: *The Journal of the British Blockchain Association* 1 (1). DOI: 10.31585/jbba-1-1-(10)2018.

Kannengießer, N., S. Lins, T. Dehling, and A. Sunyaev (2020). "Trade-offs between distributed ledger technology characteristics". In: *ACM Computing Surveys* 53 (2). DOI: 10.1145/3379463.

Kannengiesser, N., S. Lins, C. Sander, K. Winter, H. Frey, and A. Sunyaev (2021). "Challenges and common solutions in smart contract development". In: *IEEE Transactions on Software Engineering* 48 (11), pp. 4291–4318. DOI: 10.1109/tse.2021.3116808.

Karger, E. (2020). “Combining blockchain and artificial intelligence – literature review and state of the art”. In: *Proceedings of the 41st International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2020/blockchain_fintech/blockchain_fintech/6/.

Köhler, S. and M. Pizzol (2020). “Technology assessment of blockchain-based technologies in the food supply chain”. In: *Journal of Cleaner Production* 269, p. 122193. DOI: 10.1016/j.jclepro.2020.122193.

Lacity, M. C. (2018). “Addressing key challenges to making enterprise blockchain applications a reality”. In: *MIS Quarterly Executive* 17 (3), pp. 201–222. URL: https://aisel.aisnet.org/misqe/vol17/iss3/3.

Lacity, M. C. and R. van Hoek (2021). *What we’ve learned so far about blockchain for business*. URL: https://sloanreview.mit.edu/article/what-weve-learned-so-far-about-blockchain-for-business/.

Lindsay, J. R. (2020). “Demystifying the quantum threat: Infrastructure, institutions, and intelligence advantage”. In: *Security Studies* 29 (2), pp. 335–361. DOI: 10.1080/09636412.2020.1722853.

Manupati, V. K., T. Schoenherr, M. Ramkumar, S. M. Wagner, S. K. Pabba, and R. I. R. Singh (2019). “A blockchain-based approach for a multi-echelon sustainable supply chain”. In: *International Journal of Production Research* 58 (7), pp. 2222–2241. DOI: 10.1080/00207543.2019.1683248.

Marx, M., E. Zimmer, T. Mueller, M. Blochberger, and H. Federrath (2018). “Hashing of personally identifiable information is not sufficient”. In: *Sicherheit, Schutz und Zuverlässigkeit*, pp. 55–68. DOI: 10.18420/sicherheit2018_04.

Mason, E. (2021). *Bitcoin about-face: JPMorgan opens crypto trading to all clients*. URL: https://www.forbes.com/sites/emilymason/2021/07/22/bitcoin-about-face-jpmorgan-opens-crypto-trading-to-all-clients/?sh=66f2714a44a5.

Mattke, J., C. Maier, and A. Hund (2019). “How an enterprise blockchain application in the U.S. pharmaceuticals supply chain is saving lives”. In: *MIS Quarterly Executive* 18 (4), pp. 246–261. DOI: 10.17705/2msqe.00019.

McCann, C. (2000). *Detecting personal trading abuses*. URL: https://www.sec.gov/rules/other/f4-433/mccann1.htm.

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

Nofer, M., P. Gomber, O. Hinz, and D. Schiereck (2017). “Blockchain”. In: *Business & Information Systems Engineering* 59 (3), pp. 183–187. DOI: 10.1007/s12599-017-0467-3.

Oliveira, L., L. Zavolokina, I. Bauer, and G. Schwabe (2018). "To token or not to token: Tools for understanding blockchain tokens". In: *Proceedings of the 39th International Conference on Information Systems*. AIS. DOI: 10.5167/UZH-157908.

Pedersen, A. B., M. Risius, and R. Beck (2019). "A ten-step decision path to determine when to use blockchain technologies". In: *MIS Quarterly Executive* 18 (2), pp. 99–115. DOI: 10.17705/2msqe.0001.

Pezzone, J. (2022). *Intel's SGX deprecation impacts DRM and Ultra HD Blu-ray support*. URL: https://www.techspot.com/news/93006-intel-sgx-deprecation-impacts-drm-ultra-hd-blu.html.

Platt, M., R. J. Bandara, A.-E. Drăgnoiu, and S. Krishnamoorthy (2021). "Information privacy in decentralized applications". In: *Trust Models for Next-Generation Blockchain Ecosystems*. Springer, pp. 85–104. DOI: 10.1007/978-3-030-75107-4_4.

Ponciano, J. (2021). *Goldman Sachs to become second big bank offering Bitcoin to wealthy clients*. URL: https://www.forbes.com/sites/jonathanponciano/2021/03/31/goldman-sachs-to-become-second-big-bank-offering-bitcoin-to-wealthy-clients/?sh=25df6d77722f.

Queiroz, M. M., R. Telles, and S. H. Bonilla (2019). "Blockchain and supply chain management integration: A systematic review of the literature". In: *Supply Chain Management: An International Journal* 25 (2), pp. 241–254. DOI: 10.1108/scm-03-2018-0143.

Queiroz, M. M. and S. F. Wamba (2019). "Blockchain adoption challenges in supply chain: An empirical investigation of the main drivers in India and the USA". In: *International Journal of Information Management* 46, pp. 70–82. DOI: 10.1016/j.ijinfomgt.2018.11.021.

Regner, F., N. Urbach, and A. Schweizer (2019). "NFTs in practice – non-fungible tokens as core component of a blockchain-based event ticketing application". In: *Proceedings of the 39th International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2019/blockchain_fintech/blockchain_fintech/1/.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). "Building a blockchain application that complies with the EU general data protection regulation". In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Rieger, A., T. Roth, J. Sedlmeir, and G. Fridgen (2021). "The privacy challenge in the race for digital vaccination certificates". In: *Med* 2 (6), pp. 633–634. DOI: 10.1016/j.medj.2021.04.018.

Roeck, D., H. Sternberg, and E. Hofmann (2019). “Distributed ledger technology in supply chains: A transaction cost perspective”. In: *International Journal of Production Research* 58 (7), pp. 2124–2141. DOI: 10.1080/00207543.2019.1657247.

Rossi, M., C. Mueller-Bloch, J. B. Thatcher, and R. Beck (2019). “Blockchain research in information systems: Current trends and an inclusive future research agenda”. In: *Journal of the Association for Information Systems*, pp. 1388–1403. DOI: 10.17705/1jais.00571.

Saberi, S., M. Kouhizadeh, J. Sarkis, and L. Shen (2018). “Blockchain technology and its relationships to sustainable supply chain management”. In: *International Journal of Production Research* 57 (7), pp. 2117–2135. DOI: 10.1080/00207543.2018.1533261.

Schellinger, B., F. Völter, J. Sedlmeir, and N. Urbach (2022). “Yes, I do: Marrying blockchain applications with GDPR”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 4631–4640. DOI: 10.24251/hicss.2022.563.

Schlatt, V., J. Sedlmeir, S. Feulner, and N. Urbach (2022). “Designing a framework for digital KYC processes built on blockchain-based self-sovereign identity”. In: *Information & Management* 59 (7). DOI: 10.1016/j.im.2021.103553.

Schweizer, A., P. Knoll, N. Urbach, H. A. von der Gracht, and T. Hardjono (2020). “To what extent will blockchain drive the machine economy? Perspectives from a prospective study”. In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1169–1183. DOI: 10.1109/tem.2020.2979286.

Sedlmeir, J., P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021a). “The DLPS: A new framework for benchmarking blockchains”. In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

Sedlmeir, J., H. U. Buhl, G. Fridgen, and R. Keller (2020). “The energy consumption of blockchain technology: Beyond myth”. In: *Business & Information Systems Engineering* 62 (6), pp. 599–608. DOI: 10.1007/s12599-020-00656-x.

Sedlmeir, J., R. Smethurst, A. Rieger, and G. Fridgen (2021b). “Digital identities and verifiable credentials”. In: *Business & Information Systems Engineering* 63 (5), pp. 603–613. DOI: 10.1007/s12599-021-00722-y.

Sedlmeir, J., T. Wagner, E. Djerekarov, R. Green, J. Klepsch, and S. Rao (2022). “A serverless distributed ledger for enterprises”. In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 7382–7391. DOI: 10.24251/hicss.2022.886.

Shi, S., D. He, L. Li, N. Kumar, M. K. Khan, and K.-K. R. Choo (2020). “Applications of blockchain in ensuring the security and privacy of electronic health record systems: A survey”. In: *Computers & Security* 97. DOI: 10.1016/j.cose.2020.101966.

Šimunić, S., D. Bernaca, and K. Lenac (2021). “Verifiable computing applications in blockchain”. In: *IEEE Access* 9, pp. 156729–156745. DOI: 10.1109/access.2021.3129314.

Soltani, R., U. T. Nguyen, and A. An (2021). “A survey of self-sovereign identity ecosystem”. In: *Security and Communication Networks*. DOI: 10.1155/2021/8873429.

Sternberg, H. S., E. Hofmann, and D. Roeck (2020). “The struggle is real: Insights from a supply chain blockchain case”. In: *Journal of Business Logistics* 42 (1), pp. 71–87. DOI: 10.1111/jbl.12240.

Sturm, C., J. Scalanczi, S. Schönig, and S. Jablonski (2019). “A blockchain-based and resource-aware process execution engine”. In: *Future Generation Computer Systems* 100, pp. 19–34. DOI: 10.1016/j.future.2019.05.006.

Sundarakani, B., R. de Souza, M. Goh, S. M. Wagner, and S. Manikandan (2010). “Modeling carbon footprints across the supply chain”. In: *International Journal of Production Economics* 128 (1), pp. 43–50. DOI: 10.1016/j.ijpe.2010.01.018.

Sunyaev, A., N. Kannengießer, R. Beck, H. Treiblmaier, M. Lacity, J. Kranz, G. Fridgen, U. Spankowski, and A. Luckow (2021). “Token economy”. In: *Business & Information Systems Engineering* 63 (4), pp. 457–478. DOI: 10.1007/s12599-021-00684-1.

Toufaily, E., T. Zalan, and S. B. Dhaou (2021). “A framework of blockchain technology adoption: An investigation of challenges and expected value”. In: *Information & Management* 58 (3), p. 103444. DOI: 10.1016/j.im.2021.103444.

Wang, S., W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang (2019). “Decentralized autonomous organizations: Concept, model, and applications”. In: *IEEE Transactions on Computational Social Systems* 6 (5), pp. 870–878. DOI: 10.1109/tcss.2019.2938190.

Warkentin, M. and C. Orgeron (2020). “Using the security triad to assess blockchain technology in public sector applications”. In: *International Journal of Information Management* 52, p. 102090. DOI: 10.1016/j.ijinfomgt.2020.102090.

Werner, S. M., D. Perez, L. Gudgeon, A. Klages-Mundt, D. Harz, and W. J. Knottenbelt (2021). *SoK: Decentralized finance (DeFi)*. URL: https://arxiv.org/abs/2101.08778.

Whitaker, A. and R. Kräussl (2020). “Fractional equity, blockchain, and the future of creative work”. In: *Management Science* 66 (10), pp. 4594–4611. DOI: 10.1287/mnsc.2020.3633.

Whitney, K. M. and C. B. Daniels (2013). “The root cause of failure in complex IT projects: Complexity itself”. In: *Procedia Computer Science* 20, pp. 325–330. DOI: 10.1016/j.procs.2013.09.280.

Wüst, K. and A. Gervais (2018). “Do you need a blockchain?” In: *Crypto Valley Conference on Blockchain Technology*. IEEE. DOI: 10.1109/cvcbt.2018.00011.

Xu, X., H. D. Bandara, Q. Lu, I. Weber, L. Bass, and L. Zhu (2021). “A decision model for choosing patterns in blockchain-based applications”. In: *18th International Conference on Software Architecture*. IEEE, pp. 47–57. DOI: 10.1109/icsa51549.2021.00013.

Xu, X., C. Pautasso, L. Zhu, Q. Lu, and I. Weber (2018). “A pattern collection for blockchain-based applications”. In: *Proceedings of the 23rd European Conference on Pattern Languages of Programs*. ACM. DOI: 10.1145/3282308.3282312.

Zavolokina, L., R. Ziolkowski, I. Bauer, and G. Schwabe (2020). “Management, governance and value creation in a blockchain consortium”. In: *MIS Quarterly Executive* 19 (1). DOI: 10.17705/2msqe.00022.

Zetzsche, D. A., D. W. Arner, and R. P. Buckley (2020). “Decentralized finance”. In: *Journal of Financial Regulation* 6 (2), pp. 172–203. DOI: 10.1093/jfr/fjaa010.

Zhang, R., R. Xue, and L. Liu (2019). “Security and privacy on blockchain”. In: *ACM Computing Surveys* 52 (3). DOI: 10.1145/3316481.

H Research Paper 7 – Transition pathways towards design principles of self-sovereign identity

Authors:

Johannes Sedlmeir, Tom Barbereau, Jasmin Huber, Tamara Roth, & Linda Weigl

Published in:

Proceedings of the 43rd International Conference on Information Systems
(URL: https://aisel.aisnet.org/icis2022/is_implement/is_implement/4/)

Abstract:

Society's accelerating digital transformation during the COVID-19 pandemic highlighted clearly that the Internet lacks a secure, efficient, and privacy-oriented model for identity. Self-sovereign identity (SSI) aims to address core weaknesses of siloed and federated approaches to digital identity management from both users' and service providers' perspectives. SSI emerged as a niche concept in libertarian communities, and was initially strongly associated with blockchain technology. Later, when businesses and governments began to invest, it quickly evolved towards a mainstream concept. To investigate this evolution and its effects on SSI, we conduct design science research rooted in the theory of technological transition pathways. Our study identifies nine core design principles of SSI as deployed in relevant applications, and discusses associated competing political and socio-technical forces in this space. Our results shed light on SSI's key characteristics, its development pathway, and tensions in the transition between regimes of digital identity management.

Keywords:

Certificate, digital wallet, distributed ledger, innovation, public key infrastructure, verifiable credential

H.1 Introduction

According to Kim Cameron, Microsoft's former Chief Architecture of Identity, "the Internet was built without a way to know who and what [people] are connecting to" (Cameron, 2005). It typically only allows the identification of physical endpoints and the associated organizations (Tobin and Reed, 2016). End-users experience this design daily when they interact with the servers of digital service providers using an https connection (Preukschat and Reed, 2021). Servers identify themselves with cryptographic key pairs and SSL certificates, i.e., documents that are electronically signed by one of a few dozen global "certificate authorities" (Soltani et al., 2021). The resulting public key infrastructure (PKI) can thus be considered the Internet's equivalent of a public "address book" or "telephone book" for public entities, maintained by a list of reputed organizations (Adams and Lloyd, 2003). Through its integration into web browsers and mobile applications, it provides the backbone of today's trusted interactions via the Internet (Jøsang, 2014).

Despite the apparent success of digital certificates, they are rarely extended to end-users. One of the few examples include the European Union's Digital COVID certificates (Rieger et al., 2021) and the introduction of staff passports for the United Kingdom's national health service during the pandemic (Lacity and Carmel, 2022). Instead, end-user identities are typically managed through *siloed* and *federated* systems (El Maliki and Seigneur, 2007). In the siloed approach, users need to register a new account for each digital service that they interact with. Oftentimes, these accounts are just a combination of an identifier, such as a username or an e-mail address, and a credential to prove control over the identifier, such as a password or a smartcard (Whitley et al., 2014). Registering or maintaining an account may also involve filling in registration forms and visiting a company branch or government office that verifies claims such as the possession of a valid driver's license (Sedlmeir et al., 2021). Resulting records can be verified by the digital service provider and stored on its servers, so simplifying future verification processes. However, manual registration and the secure management of passwords for sometimes hundreds of digital services presents a substantial challenge and inconvenience to end-users (Bonneau et al., 2012). Related challenges for companies and governments lie in maintaining security, supporting operations, and manually verifying users' attributes (Schlatt et al., 2022; Smith and McKeen, 2011).

To address these downsides, dedicated identity providers (IdPs) entered the market (Maler and Reed, 2008). Examples for IdPs are companies like Google and Microsoft and government agencies like the Unique Identification Authority of India (Sedlmeir et al., 2021).

As in the siloed approach, IdPs store (and to some extent verify) their users' identity attributes. Additionally, they enable users to authenticate with other service providers that connect with the IdP using their IdP account. Technically, when logging in to a digital service, users are redirected to their IdP, where they sign in with their corresponding credential. The IdP then forwards an attestation of the required identity attributes to the service provider (Madsen et al., 2005; Maler and Reed, 2008). As the resulting network of IdPs and digital service providers resembles a federation, this identity paradigm is called federated identity management (Maler and Reed, 2008). While the "single sign-on" experience of the federated approach is efficient and convenient for users, it is often criticized for the centralized storage of identity data and corresponding cyber-security risks and surveillance risks. Moreover, IdPs often monetize their users' identity and usage data (van Bokkem et al., 2019; Zuboff, 2015), taking powerful market positions. Federated identity management also has not yet addressed the lack of machine-verifiable digital representations of core identity-related documents such as passports, driver's licenses, or diplomas (Sedlmeir et al., 2021).

The shortcomings of the siloed and federated approaches have led to growing interest in a *user-centric* and *decentralized* digital identity paradigm (El Maliki and Seigneur, 2007; Kubach et al., 2020; OECD, 2011). Attempts to implement this paradigm in the context of e-commerce and enterprise IT systems date back to the early 2000s (Backes et al., 2005; Chadwick et al., 2003). These endeavors have ultimately led to the concept of self-sovereign identity (SSI) – an expression of personal digital sovereignty. It emerged as a "technological niche" (Geels, 2004) among digital identity communities, most notably, the Internet Identity Workshops (IIWs), which previously played a major role in the development of federated identity standards (Preukschat and Reed, 2021). Subsequently, Allen (2016), who was a leading figure in incubating SSI, coined the term as a principle-based framework for a decentralized system of user-centric digital identities. His "10 principles of SSI" provide the first definition of SSI. At that time, there were no relevant reference standards or practical experiences with the large-scale deployment of SSI-based systems and their interaction with the regulatory, technical, and economic environment. Since then, through inter- and intra-organizational proofs of concept and pilot projects in businesses and public services, SSI has evolved considerably (Schellinger et al., 2022). Different technological components of SSI and various identification and authentication scenarios were explored (Sedlmeir et al., 2021; Soltani et al., 2021). However, the development of guidelines and design considerations for SSI system implementation or evaluation has stalled or, at best, evolved in heterogeneous directions based on no or weak scientific evidence. For instance, Allen's principles stem from a blog post and mainly focus on libertarian values like autonomy and privacy; yet, applications of SSI in industry and e-government also require specific authenticity and accountability guarantees (Kubach et al., 2020). Moreover, regulatory aspects like the different "levels of assurance" formulated in the European electronic Identification, Authentication, and Trust Services (eIDAS) regulation impact practical SSI implementations (Schellinger et al., 2022; Schwalm et al., 2022). The continuous innovation and evolution process within the SSI community hence cannot be viewed merely from a techno-centric perspective. Indeed, the concepts of "sovereignty" and "decentralization" in the context of digital identity are contested (Sedlmeir et al., 2021) and subject to different interpretations according to actors' social and institutional context (Weigl et al., 2022). Consequently, SSI-solutions should be understood and analyzed as innovations with "political-economic dimensions" (Dijck and Jacobs, 2020).

Related research on SSI is scarce and has not captured this context thus far. As a result, "SSI is still only loosely defined" (Mühle et al., 2018) and there seems to be no updated definition of SSI that includes both practitioners' and researchers' perspectives. The academic debate on SSI is also fuzzy: while initially scholarship emphasized the role of blockchain as an essential technological building block (e.g., Koens and Meijer, 2018; Mühle et al., 2018), more recent research suggests a smaller role for blockchain (Schlatt et al., 2022). In the last years, there has been a noticeable trend towards, among others, a stronger focus on applications in regulated domains, user experience, privacy-oriented implementations, and the bundling of attestations (Feulner et al., 2022; Sartor et al., 2022; Schwalm et al., 2022; Soltani et al., 2021). Harmonized design principles (DPs) are required for research and practice, e.g., to evaluate identity management concepts and solutions consistently and not only from a techno-centric and deductive perspective (e.g., see Koens and Meijer (2018)). Considering the diversity of technical niche innovations, socio-technical developments, and the influence of an exogenous landscape which impacted the adoption of SSI, we believe that a rigorous and timely assessment of the key characteristics of SSI is required. We provide an updated model in the form of DPs for SSI that supplements the libertarian concept as introduced by Allen (2016) with influences of the technical environment as well as regulatory and business requirements in terms of accountability, authenticity, and trust structures.

To derive these principles, we use the multi-level perspective (MLP) by Geels and Schot (2007) as a theoretical lens to retrace the *transition pathway* of SSI from a technological niche towards a mainstream concept. Through this theoretical lens, we derive the DPs following a design science research (DSR) study (Hevner et al., 2004; Peffers et al., 2007). We introduce Geels and Schot's MLP and use it to give a first, informal overview of different SSI-related historical milestones and evolutions in identity management. They illustrate the complexity of technical foundations and paths involved, and highlight the need for multi-faceted research to formally structure and map these developments (Whitley et al., 2014). Next, we present our DSR, which involves a systematic literature review (SLR) to develop the initial version of DPs for SSI and four subsequent iterative refinement and evaluation cycles in which we interview 15 experts from academia and businesses on SSI. We then discuss the implications of the developed DPs for the area of SSI, especially in the context of Allen's principles. We also point to related tensions that we observed in SSI's transition from being principally a libertarian theoretical construct to a practical identity management paradigm. Finally, we summarize our findings and outline the need for further developments and research in the area of SSI.

H.2 Background

Digital identity management models can be viewed as socio-technical constructs undergoing a permanent process of innovation (Seltsikas and O'Keefe, 2010; Smith and McKeen, 2011; Whitley et al., 2014). Leaning on science and technology studies (STS), questions pertaining to technology development build on theories of technological entrenchment and strategies to incubate or sustain novel technologies. The concept of entrenchment stems from the idea that "when change is easy, the need for it cannot be foreseen; [though] when the need for change is apparent, change has become expensive, difficult, and time-consuming" (Collingridge, 1980). That is, the convenience of an established solution, called the "entrenched" solution, makes change difficult to achieve as neither social nor economic or political drivers for change exist (Geels, 2002). Over the past 40 years, numerous researchers have analyzed this phenomenon in the context of technological innovations (e.g., Callon, 1986; Hughes, 1983). They assume innovation takes place in protected niches where technologists safely develop and improve their technology, which – over time – "stabilizes as the outcome of successive learning processes" to form new regimes (Geels, 2004).

The multi-level perspective (MLP) was introduced as part of STS and dissects the innovation process in terms of 'technological niches", the established "socio-technical regime", and the larger "exogenous landscape" (Geels, 2004). Respectively, the framework consists of three levels — the micro, meso, and macro level – upon which different selec-

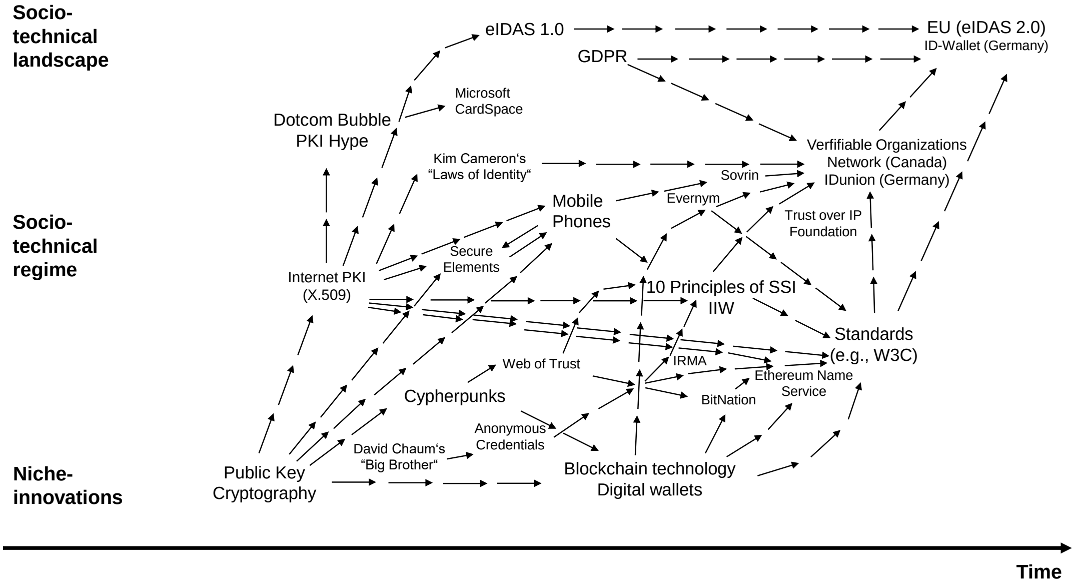


Figure 1: Multilevel perspective on selected key events and their interdependencies in identity management.

tion factors apply to drive innovation and shape technology development. Technological niches construct the framework's micro-level. At this level, radical novelties emerge, that is, innovations deviating considerably from the existing regime. Established regimes reside at the meso level and are often characterized by lock-in and path-dependent mechanisms of economic, social, organizational, or political nature (Geels, 2002). Lastly, the macro level contains the wider exogenous landscape in terms of the socio-political and economic conditions that may change and create "windows of opportunity" through which niche innovations can emerge (Geels, 2004; Geels and Schot, 2007). We aim to use the MLP as a theoretical lens to consolidate and contextualize the phenomenon of SSI-based identity management. Moreover, our work contributes to the stream of Information Systems research that explores technical opportunities and policy recommendations as well as more general managerial and societal questions associated with the development of identification technologies (Sedlmeir et al., 2021; Whitley et al., 2014). Prior to doing so, the development of SSI ought to be contextualized within past regimes. Hence, by adopting the MLP, Figure 1 structures the key events and their influences on the evolution of SSI that we present in the following.

Public key cryptography can be considered the most foundational part of both the existing trust layer on the Internet and implementations of SSI. While originally invented by Ellis and Cocks in 1973/74, the first publication by Rivest et al. (1978) resulted in an instantiation of the eponymous RSA cryptosystem. Public key cryptography uses one-way functions to derive a public key – typically a large number that can be considered a non-human-readable identifier – from a randomly generated secret key. The ownership of the key pair, i.e., knowledge of the secret key, can be proven mathematically without disclosing the secret key itself. The mathematical connection between the secret key as credential and the public key as identifier also opens up new opportunities for digital identity management beyond mere authentication. When it comes to presenting identity attributes for the purpose of identification or authorization, these can be verifiably claimed through digital certificates. That is, an "issuer" – either a reputed person or an organization known by its public key – uses its own secret key to electronically sign a document that lists the subject's public "binding" key along with its other identity attributes. An identity subject can then send this digital certificate and a proof of ownership of the binding key in a *verifiable presentation* directly to a relying ("verifying") party, for instance, to a service provider. The latter can cryptographically check the integrity of this digital certificate based on the issuer's digital signature. Provided that the verifying party trusts the issuer, it can then rely on the attested attributes. In the context of institutions and their digital services, this has evolved into today's system of X.509 certificates for servers and the Internet's PKI (Chadwick et al., 2003). Within the MLP, we understand PKI standards and related infrastructural components as a socio-technical regime that received significant adoption with the Dotcom bubble, became stable, and remained widespread through its crucial role for https-based communication.

"Cypherpunks" is the name given to libertarian and privacy-oriented communities that make use of cryptographic tools to pursue their goals (Narayanan, 2013). Some of these groups made early attempts to create a "Web of Trust" using cryptographic key pairs and digital certificates, issued by end-users for end-users (Zimmermann, 1995). An example of this is the implementation of "Pretty Good Privacy". In the early 2000s, attempts were made to base these efforts on institutional trust instead of social trust. A key goal was to improve digital identity management in areas such as e-commerce or enterprise IT by extending the Internet's PKI for organizations and their servers to use by individuals. They used, for instance, smartcards that securely store key pairs and certificates issued by the users' employers (Chadwick et al., 2003). While the vision to extend this user-centric and cryptography-oriented approach failed to gain large-scale traction, it prevailed for some time in niche communities. This mostly included computer scientists and cypherpunks who took seriously Chaum's warnings of surveillance threats on the Internet and corresponding spillover effects on society (Chaum, 1985, "Big Brother"). They explored cryptographic tools to minimize information exposure during a verifiable presentation. In cryptography research, this led to innovative solutions. In contrast to established digital certificates, anonymous credentials (also called attribute-based credentials) facilitate zero-knowledge proofs to provide data-minimal evidence on the ownership of a digital certificate and required attributes. That is, an anonymous credential allows to derive verifiable presentations without revealing all the attributes that it attests. It also allows to avoid the disclosure of an associated unique identifier, such as the binding public key or the value of the issuer's digital signature (Backes et al., 2005; Camenisch and Lysyanskaya, 2001). IRMA ("I Reveal My Attributes") was one of the first practical implementations of these anonymous credentials (Alpár and Jacobs, 2013). Besides privacy, niche innovations also emerged in communities of cryptographers and cypherpunks who sought to minimize the involvement of trusted third parties like certificate authorities. After Bitcoin and blockchain technologies gained a broader foothold, actors driven by libertarian values saw opportunities to establish a registry for digital identities by mapping individuals to their public keys on a transnational digital infrastructure. This rekindled interest in using public key cryptography for end-users' identity management resulted in projects like BitNation (Kuperberg, 2019). In addition, the popularity of tools to manage cryptocurrencies made citizens and decision-makers in industry and politics aware of the opportunities of identity management via digital wallets applications on smartphones (Jørgensen and Beck, 2022; Sartor et al., 2022).

The term SSI was coined by Allen (2016) in a blog post. His "principles of SSI" encompass users' independent *existence* (1); the *control* (2) they must have over their identities; the *access* (3) users are granted to their own data; the *transparency* (4) of related systems and algorithms' implementation; the *persistence* (5) of identities for as long as users wish; the *portability* (6) of attestations tied to users' identities; *interoperability* (7); *consent*-based (8) sharing of users' identity data; privacy through disclosure *minimalization* (9); and, finally, users' rights *protection* (10). The concept has since become a focal topic far beyond the relatively narrow focus of the half-yearly IIW conferences (Čučko and Turkanović, 2021; Soltani et al., 2021). While gathering "internal momentum" (Geels and Schot, 2007), the principles stipulated within this group soon became reference points for SSI solutions. In parallel, the first blockchain-based implementations of SSI appeared, such as Evernym's solution based on what later became Hyperledger Indy and Aries. Their efforts significantly influenced technical and non-technical standards, which were refined from a governance perspective, for instance, by Sovrin and the Trust over IP foundation and from a technical perspective by the World Wide Web Consortium (W3C) and the Decentralized Identity Foundation. Arguably, the two most important standards in the context of SSI are “decentralized identifiers” – public keys enriched with meta-data – and “verifiable credentials” – digitally signed attestations that offer higher flexibility with regard to semantics and that enable them to incorporate meta-data and features of anonymous credentials (Sedlmeir et al., 2021). Within these smaller regimes, respective socio-technical configurations for SSI were established.

The configurations in individual regimes, however, are not homogeneous. Instead, they can be considered “sequences of multiple component-innovations” (Geels and Schot, 2007) that are continuously reconfigured and converge into a solution. The heterogeneity in configurations manifests itself, for instance, in the contested use of blockchain as a component. The realization that pseudonymous public keys do not provide sufficient privacy (Sedlmeir et al., 2022), and that the immutability of a blockchain is not required for digital attestations signed by an issuer (Schlatt et al., 2022), diminished the role of blockchain in more recent SSI implementations. In many projects, end-users’ identifiers, endpoints, and attestations are now exclusively stored in digital wallets on their devices. A blockchain then at most hosts the PKI for public institutions as well as revocation registries (Lacity, 2022; Schlatt et al., 2022). This can be seen, for instance, in Canada’s Verifiable Organizations Network, the European cooperative society IDunion, and the European Self-Sovereign Identity Framework’s technical approaches. SSI projects are often tied to dynamics in the socio-technical landscape. Ongoing political initiatives, like the revision of the European eIDAS regulation and the desire to establish a German ID Wallet, manifest the attention SSI has obtained from the regulatory domain. The development of SSI for identity management hence reflects the interplay of the MLP’s different levels and the corresponding technical, socio-economical, and political selection factors. SSI is often hailed as a revolutionary innovation, yet its implementations are not considerably different from early proposals of using PKI and anonymous credentials stored on end users’ portable computing devices (Backes et al., 2005; Chadwick et al., 2003). Arguably, public key cryptography alone contributes significantly to more secure and efficient identity management (Bonneau et al., 2012). Blockchain technology, which is still a component of many instantiations of SSI, only plays a minor role from a technical perspective (Schlatt et al., 2022). Yet, it appears to have contributed to its initial broad-based hype, as previous moderate attempts to lobby for the adoption of public key cryptography and digital certificates by end-users in research (e.g., Rannenberg et al., 2015) and policy (e.g., eIDAS) have not received the anticipated widespread adoption (Kubach et al., 2020). This mirrors Geels (2004)’s proposition that despite technical superiority over the incumbent technical solution, other factors beyond the technological regime influence successful adoption of a new regime. Since SSI connected with blockchain technology, there has been somewhat unprecedented support from political decision makers (Weigl et al., 2022).

H.3 Research approach

For our DSR approach, we first identified the problem space to obtain descriptive knowledge on SSI solutions that researchers currently discuss through an initial SLR (Gregor and Hevner, 2013; vom Brocke et al., 2020). We then gathered qualitative data from the SLR and subsequent 15 expert interviews (Sonnenberg and vom Brocke, 2012). During data collection, we challenged, validated, and refined our tentative results against current practices and discussion in IT development and industry in iterative rephrase-and-evaluate loops (Gregor and Hevner, 2013; Hevner et al., 2004; Peffers et al., 2007). In this process, the MLP allowed us to contextualize our findings from the SLR on the various characteristics of SSI and the trajectories of its technical constituents. To integrate existent design knowledge into our endeavor to create additional, generalizable design knowledge (vom Brocke et al., 2020), we focused on the present solution space of SSI. More specifically, we reviewed and consolidated existing DPs from literature and SSI projects in a DSR study to derive DPs for SSI as a form of decentralized digital identity management. As related developments are driven by both theory and practice (Allen, 2016; Camenisch and Lysyanskaya, 2001; Preukschat and Reed, 2021; Whitley et al., 2014), DSR allowed us to consolidate observations from either perspective. A first set of DPs typically builds on Ω-knowledge or descriptive knowledge, which conveys an understanding of the laws and regularities of an observed phenomenon. Subsequent evaluation and sense-making processes then help derive a finite set of DPs, commonly referred to as Λ-knowledge or prescriptive knowledge (Gregor and Hevner, 2013; vom Brocke et al., 2020). According to the knowledge contribution framework, our DSR approach follows the precept of *exaptation*. Exaptation requires the extension of a known solution to new problems (Gregor and Hevner, 2013). Digital identity management is a well-known research topic (Smith and McKeen, 2011; Whitley et al., 2014) and often makes use of cryptographic components. Yet, the challenges we identified in the Introduction section have necessitated a paradigm shift. Current design knowledge, however, is often too unspecific and applications too versatile to derive generally accepted DPs for SSI (Preukschat and Reed, 2021). To address this problem, we consolidate existing and extend current design knowledge in generalizable and actionable DPs (Gregor and Hevner, 2013).

In line with Webster and Watson (2002) and Fink (2019), we extracted 2,504 publications from 14 databases, including ACM DL, IEEE Xplore, ScienceDirect, Scopus, Springer Link, Web of Science, and Google Scholar for our SLR. We started with two initial search strings, “self-sovereign identity” and “self-sovereignty”, to get an overview of current research on SSI. We used the initial results to extract additional relevant keywords that had not yet been included in our search string. Owing to the close connection between blockchain and SSI communities as discussed in the Background section, our final search string then comprised keywords from the identity and blockchain realm: “self-sovereign identity” OR self-sovereignty OR (identity AND (blockchain OR decentrali*ed)). The term “decentralized”, as influenced by Kuperberg (2019), seems an essential characteristic of SSI and inextricably linked to the concept, also through its strong link to blockchain communities (Weigl et al., 2022). In a title screening, we identified 84 publications as potentially being relevant. After a detailed full-text analysis of these contributions and applying inclusion (detailed discussion or use of design or evaluation criteria for SSI systems) and exclusion criteria (no English language, article not accessible, purely cryptographic content), 14 publications remained. A subsequent forward and backward search (Fink, 2019; Webster and Watson, 2002) yielded another 8 publications, seven of which are gray literature, technical standards (e.g., by the W3C), or laws (the EU’s general data protection regulation (GDPR)). Yet, two of the most popular contributions on SSI (Allen (2016) and Cameron (2005)) could not be extracted with our SLR, as they represent blog posts that are typically not listed in academic databases. We included these two contributions in our knowledge base since they contain essential definitions of SSI and discussions about key requirements.

Our approach towards DPs for SSI-based digital identity management follows the two modes of “kernel theory to design entity grounding” and “design entity to design theory grounding” to enrich the current knowledge base (vom Brocke et al., 2020). The evaluation of various approaches to implement SSI based on our SLR in combination with information retrieved from the basket of literature and projects on identity management referenced in the Introduction and Background sections helped us to derive design requirements. These served as solution fitness criteria for the challenges of digital identity management from the perspective of end-users, businesses, and regulators. Evaluations of existing approaches additionally delivered design features that we included in the development of a first set of DPs (Gregor and Hevner, 2013; vom Brocke et al., 2020). To increase their projectability, we evaluated and complemented them in four iterative evaluation cycles. The outcome was a nascent design theory in the form of a consolidated set of DPs (Hevner et al., 2004; Peffers et al., 2007; vom Brocke et al., 2020). Throughout this iterative process, we followed the suggested procedure of Hevner et al. (2004) to refine the DPs in 15 evaluation interviews with six researchers and nine industry experts, who are all highly esteemed in the field of SSI design and implementation. The practitioners represent relevant organizations and projects from niche innovations and the socio-technical regime (some have multiple of the following roles): Five interviewees have been regular attendees and presenters at last years' IIWs, and eight of them are actively involved in SSI-related standardization bodies like Sovrin, the Trust over IP foundation, and the W3C. Two interviewees are among the four editors of the W3C decentralized identifiers standard, which is also co-authored by Christopher Allen. Five interviewees are in leading positions for the implementation of the Verifiable Organizations Network or the IDunion project within their company, and four of them represent businesses that develop cloud and edge SSI wallets in Europe and North America. Moreover, we communicated our findings beyond exchanging ideas in the expert interviews as recommended for the DSR (Hevner et al., 2004). This included presentations of our work at the IIW, where it served as a discussion basis for the Principles of SSI, which were later – including adjustments – published by the Sovrin Foundation (2021). This work also considerably influenced a related compilation by the Trust over IP Foundation (2021). The aim of the interviews was to ensure the parsimony of our DPs for the creation of SSI-based solutions. To achieve parsimony, we controlled for the completeness, usefulness, and understandability of our DPs throughout the interviews. Interviewees were each encouraged to review the entire list of DPs and to provide (1) additions to the list, (2) reframing of existing DPs, and (3) changes to the definition of DPs. We also discussed openly the current state of decentralized digital identity management as well as the technical and social foundations, opportunities, and challenges of these approaches as perceived by the interviewees. The semi-structured interviews hence allowed the interviewees to elaborate on their professional perspective of SSI. We conducted each interview remotely. The interviews lasted between 30 and 60 minutes and were audio-recorded and transcribed afterwards. We refrained from scheduling new interviews once we reached a point where the interviewees provided us with almost identical feedback and did not suggest any further additions (Myers and Newman, 2007). For both the coding of selected literature and the interviews, we performed a two-stage process of inductive and deductive coding, as recommended by Miles et al. (2018). That is, two authors first separately analyzed the data, assigning codes to identify factors relevant to the design of SSI applications. They then abstracted these codes into higher-level concepts, i.e., our first tentative DPs from literature (deductive coding) and their refinement during the analysis of the interviews (inductive coding). After the literature coding and every fifth interview, the independent authors compared and discussed their results where diverging (Miles et al., 2018).

We connected the DPs with our kernel theory, the MLP, by discussing them against the backdrop of SSI's trajectory through the socio-political landscape and its interaction with legacy systems. This should ensure the relevance of our DPs (Hevner et al., 2004; Peffers et al., 2018) and, moreover, demonstrate that SSI as a form of decentralized digital identity management has developed from a radical niche to an acknowledged design (Geels, 2004; Geels and Schot, 2007) in private- and public-sector applications (Schlatt et al., 2022; Soltani et al., 2021). That is, our nascent design theory can be categorized as a design relevant explanatory or predictive theory. Our DPs enrich theories that have been relevant to initial design choices (Kuechler and Vaishnavi, 2012) such as those defined by Allen (2016). Our discussion of the resulting DPs through the lens of MLP additionally epitomizes the ascendance of technologies into broad-based adoption and provides an outlook for how SSI could further develop (Geels, 2004; Geels and Schot, 2007).

H.4 Findings

In the SLR coding process, we focused on identifying design requirements and design features for SSI management systems. While both design requirements and design features are often broad, they provide the basis for the formulation of DPs (Hevner et al., 2004; vom Brocke et al., 2020). Some requirements within the literature are already formulated as DPs (e.g., Allen (2016) and Tobin and Reed (2016)) but – dependent on their definition and relative position in the history of SSI development – may only cover a fraction of what may be relevant to date. We clustered these design requirements and features into a first set of nine DPs. In the following evaluation rounds, we added and removed one DP and adapted the remaining DPs until we reached a point where three subsequent interviews did not propose any meaningful changes. We first present the tentative DPs compiled on the basis of the SLR, and subsequently describe the changes implemented during the refinement cycles.

H.5 From design requirements and features to tentative design principles

DP1: Human Replicate. To account for the target group of SSI-based digital identities, the design requirements "human integration" (Cameron, 2005) and "human requirements [in the form of] privacy [and] empowerment" (Goodell and Aste, 2019) as well as the design feature "biometric interfaces" (Koens and Meijer, 2018) show a clear focus of SSI on natural persons, who seek to play a more active role in the management of their identity-related data. The features "reliable credential management" (Grüner et al., 2019), "data ownership", "data control", "consent to data processing" (Ferdous et al., 2019), and "portability of data" (Tobin and Reed, 2016) further emphasize the purpose of SSI as a collection of attributes related to a natural person. These can be kept for a person's entire life and, upon display, be used to disclose identity attributes. Thus, SSI enables increased agency and independence for natural persons, who wish to manage access to and distribution of their personal data. An identity considered as "self-sovereign" hence needs to be understood as collection of attributes of a real existing human being, but only of the parts they are willing to show – also called partial identities (Clauß and Köhntopp, 2001). Moreover, Abdullah et al. (2019) emphasize the concept of guardianship to give all individuals equal access to using an SSI.

DP2: Control. The design requirement of "deciding on the displayed information" (Ferdous et al., 2019) grants users of SSI "data control" (e.g., Alsayed Kassem et al., 2019; Whitley, 2009; Windley, 2019). How and when their data is being used warrants their explicit "consent to data processing" (Allen, 2016; Alsayed Kassem et al., 2019; Cameron, 2005; Ferdous et al., 2019). Controlling hence limits "what personal data is made available to others" (Whitley, 2009). This also includes the design feature of "updateability" and "revocability of consent" (Moe and Thwe, 2019) and is directly linked to the proposed identity life cycle of Koens and Meijer (2018), which contains the design features "create, attest, show, prove, renew, delete, and revoke". As such, SSI involves not only consent and control when sharing identity-related information but also "availability", i.e., the identity subject's ability to access and share verifiable information anywhere and at any time (Ferdous et al., 2019). Yet, in the context of verifiability, this does not mean that users should be able to modify all their identity information according to their liking.

DP3: Flexibility. To share their data anywhere and at any time, user-centric applications of SSI need to consider the design features "standardization" and "interoperability" (Allen, 2016; Ferdous et al., 2019; Tobin and Reed, 2016) among the different digital identity management solutions. The feature "pluralism of operators and technologies" (Cameron, 2005) should not hamper the feature "integration" (Kuperberg, 2019) of the various approaches to fulfill the design requirement of a "consistent experience across contexts" (Cameron, 2005). This also includes the design feature "portability of data" (Abraham, 2017; Allen, 2016; Ferdous et al., 2019; Tobin and Reed, 2016) in the form of identity attributes and corresponding attestations to other providers. That is, users should be able to decide which implementation to build upon – including a choice of their digital wallet. They should be empowered to consider their needs, independent of providers, and should be guaranteed interoperability with underlying technical and semantic standards.

DP4: Security. Aside from interoperability and standards, SSI-based solutions must also guarantee for the design requirement "confidentiality" which – besides availability and integrity – constitutes security. It not only entails the design features of "protection" from data accumulation, data fraud, and more powerful entities (Allen, 2016; Tobin and Reed, 2016) but also the limitation of storage and use of information for non-specified purposes as demanded by the GDPR. Overall, users should be protected from unwittingly or mistakenly sharing information with third parties, thus providing "end-to-end security" (Cavoukian, 2009). This includes also purely bilateral communication, end-to-end encryption (Goodell and Aste, 2019), and the verification of the involved verifying party's identity in a verifiable presentation to avoid man-in-the-middle attacks (Toth and Anderson-Priddy, 2019).

DP5: Privacy. Closely related to security is user privacy. In the context of SSI, it generally refers to the minimal disclosure of information, which provides users control over the degree of anonymity in interactions based on the support for unique pairwise pseudonyms for each individual private connection. Relevant design requirements and design features either directly demand "privacy by design and by default" (Cavoukian, 2009) and a high level of "pseudonymity" via pairwise unique digital identities and public keys as well as "private agents" with no storage of private data on the underlying ledger (Alsayed Kassem et al., 2019; Moe and Thwe, 2019; Windley, 2019). This allows to ensure the "unobservability" and "unlikability" (Moe and Thwe, 2019) of user information, if required. Moreover, "selective disclosure" serves as a design feature to reveal only the identity attributes relevant for a specific interaction and purpose (Cameron, 2005; Ferdous et al., 2019; Windley, 2019). Anonymous credentials (Soltani et al., 2018) and zero-knowledge proofs (Stokkink and Pouwelse, 2018; van Bokkem et al., 2019) are often mentioned as technical backbone for such enhanced privacy design features.

DP6: Credibility. Despite the goal of privacy protection, information should be authentic and verifiable also regarding timeliness. This includes the opportunity to revoke attestations from the side of the user in the case of loss or theft of the digital wallet, or or from issuers' side to account for changes of attributes and authorizations (Mühle et al., 2018). One way of implementing these design features without the need to interact with the issuer in a verifiable presentation is through the support for expiration dates and the use of revocation registries (Mühle et al., 2018). Credibility also reflects the design requirements of “transparency” (Abraham, 2017; Allen, 2016; Tobin and Reed, 2016) as well as the design features of “disclosure” (Ferdous et al., 2019), “identity assurance” and “identity verification” (Toth and Anderson-Priddy, 2019).

DP7: Authenticity. Only the respective subject should be able to pass on their data to requesting third parties. Pseudonym or credential sharing among different users, or the creation of new credentials by combining ones that do not belong to a single individual, should not be possible. Such systems exhibit “consistency of credentials”, which can, for instance, be achieved through biometric interfaces and hardware-bound link secrets or be disincentivized by corresponding PKI-assured economic bonds or all-or-nothing non-transferability (Camenisch and Lysyanskaya, 2001; Hardman, 2019). If transactions break general laws or credentials are used in an unauthorized way, global or local anonymity revocation may be useful (Camenisch and Lysyanskaya, 2001; Koens and Meijer, 2018).

DP8: Usability and Performance. Aside from verification and authentication mechanisms as the very core of SSI-based solutions, general concepts of usability must be considered to fulfil the design requirement of “user empowerment” (Abraham, 2017; Alsayed Kassem et al., 2019; Goodell and Aste, 2019). A related requirement, “positive end-user experience” (Kuperberg et al., 2019), plays a major role in delivering other requirements, such as “user trust” – which is essential for acceptance (Seltsikas and O’Keefe, 2010) – and “self-sovereign digital identity management” (Yan et al., 2017). While the “positive end-user experience” mainly complements the design feature of “user-friendly interfaces”, it may also concern features such as “scalability” (Koens and Meijer, 2018), “minimum downtime”, and “efficient performance” (Camenisch and Lysyanskaya, 2001; Kuperberg et al., 2019). Thus, SSI-based digital identity management approaches require intuitive and easy access personal data, as well as the streamlined and quick sharing of information.

DP9: Future orientation. In addition, the success of SSI largely depends on how well it fits the surrounding environment (Kuperberg et al., 2019). To enable such a fit, there are a number of economic design requirements, including the “prevention of monopolization” as well as “empowerment of businesses” (Goodell and Aste, 2019) and “manageable costs” (Ferdous et al., 2019). These requirements rely heavily on design requirements such as “efficient protocols” (Camenisch and Lysyanskaya, 2001), “organizational flexibility” and “local storage” (Abraham, 2017) as well as design features such as “decentralized governance" (Ferdous et al., 2019; Windley, 2019). Thus, we conclude that SSI-based digital identity management approaches need an innovative environment that allows structural changes to implement SSI, including adaptations of governance and agile management.

H.6 Design iterations

From the first to the second design iteration, we removed the specification of "Human" before the first tentative principle Human Replicate (TDP1). We did this because according to Expert 2 (Practitioner), smart devices and organizations can also use an SSI. Regarding Control (TDP2; DP2), Experts 1 (Researcher) and 2(P) detected potential tensions between increased control (i.e., user empowerment) and an undesirable amount of responsibility that "people now are not used to having". Open-source licensing agreements and legal compliance may be additional determining factors of Flexibility (TDP3; DP3). This was also closely linked to criticism on Credibility (TDP6) and Authenticity (TDP7), which would currently neglect the "rules of trust and basically Web of Trust, where you have to make sure the data coming from the issuer is credible" (Expert 2(P)). Experts 1(R) and 2(P) generally regarded "performance [to be] a subtopic of usability" (TDP8) and both as non-functional requirements instead of a DP, so we adjusted our TDP8 on Usability and Performance accordingly. Regarding Future orientation (TDP9), Expert 2(P) missed "bridging the gap between self-sovereign identity and the existing world of authentication and authorization" to create functional SSI.

From the second to the third design iteration, Security (TDP4; DP5) and Privacy (TDP5; DP5) were highlighted as particularly relevant (Experts 4(P), 6(R)), while the adjusted Usability (TDP8) still appeared to be deficient, neglecting other "important usability factors", such as "ease of use" and literacy, as well as the simplicity of information access. Expert 4(P) considered Future orientation (TDP9) as important, yet more of a requirement than a principle. It would indirectly already be represented in several other DPs, such as Control (TDP2) and Flexibility (TDP3). For Credibility (TDP6), the focus on revocability of consent was too narrow ("revoke the credential if it is a fake passport or whatever"), which is why we took the more general term "revocability" to also account for revocation due to incorrect data. Moreover, we renamed the previously iterated TDP1 Replicate to Representation (DP1), as the term Replicate may be uncommon and difficult to understand.

Principle	Description (key features)
DP1: Representation	SSI can represent any entity digitally – human, legal, or technical. (Attributes, authentication, existence, identification, partial identities, persistence)
DP2: Control	Only the actual controller has decision-making power over their digital identity. (Access, manage, ownership, right to be forgotten, single source of truth, update)
DP3: Flexibility	No vendor lock-in: low switching costs, focus on interoperable standards, and open-source projects. (Documentation, integration, no monopoly, portability, standards, transparency)
DP4: Security	State-of-the-art cryptographic tools and authenticated, end-to-end encrypted interactions. (Identification of relying party, key management, protection, secure communication, tamper-proofness)
DP5: Privacy	In each interaction, only the data that is essential for its purpose is revealed. (Bilateral by default, consent, minimized correlation, need to know, selective disclosure)
DP6: Verifiability	The validity and timeliness of credentials can be checked efficiently. (Certificate chain, credential management, machine readability, provability, revocability)
DP7: Authenticity	Credentials are bonded to their initial bearers. (Binding, consistency of credentials, identity fraud protection, limited transferability, risk-based authentication)
DP8: Reliability	There is guidance that helps verifiers to decide which issuers they can trust in a highly dependable infrastructure. (Decentralization, governance, guidance, no single point of failure, public registration, scalability, Web of Trust)
DP9: Usability	Success and durability factors. (Efficiency, end-user experience, minimum downtime, multiple access points, performance, recovery, simplicity, support)

Table 1: Final design principles and their definitions, including key features for implementation.

From the third to the fourth design iteration, we eliminated Future orientation (TDP9). This is because the experts considered an environment with both innovative and legacy features to be more a basic requirement than a DP specific for the implementation of SSI. As the interviewees considered the term of DP1 to be a subset of the principle alongside authentication – "because it is everything, like identification, authentication, and that you exist" (Expert 6(R)) – we renamed and redefined the DP. Regarding Flexibility (TDP3), Experts 5(P) and 11(P) suggested renaming it "openness". We refrained from doing so as it would neglect other essential properties of the principle such as interoperability and portability. In accordance with interview feedback, which offered criticism that it was "too specific" and did not include "more general points" (Expert 9(R)), we redefined Privacy (TDP5). Experts 2(P), 5(P), and 6(R) also suggested redefining Credibility (DP6), as they considered it to be too focused on technological building blocks that yet have to be established. We refrained from adding "decentralization" as a separate DP as it is a basic "prerequisite of the infrastructure" (Expert 5(P)) but added it to Future orientation (TDP9). Moreover, we renamed Credibility (TDP6) to Verifiability (DP6) and redefined Authenticity (DP7).

During the fourth design iteration – which yielded the final and consolidated set of DPs – we received positive feedback from our Experts 13(P), 14(R), and 15(R). In accordance with their feedback, we summarized the current definitions within the most relevant and generalizable core statement and exchanged the order of Usability (TDP8) and Reliability (TDP9) to Usability (DP9) and Reliability (DP8) in line with their perceived importance. Table 1 features the final DPs, including a subset of terms often used in related work and by the interviewees. The DPs characterize SSI as a user-centric "identification infrastructure" (Whitley et al., 2014) based on cryptographically verifiable attestations not only for organizations and their servers but also for end-users, maintained and controlled in digital wallets on their mobile devices (Sedlmeir et al., 2021; Soltani et al., 2021).

H.7 Discussion

The derivation of DPs delivered theoretical insights into how to develop design knowledge from such broad-based technological innovations using DSR. At first glance, our derived DPs are similar to the "Ten Principles of SSI" by Allen (2016). When Allen conceived these, SSI was mainly a theoretical concept and a formulation of key characteristics of an identity management that neither had a foundation for technical implementation, nor a history of real-world use. Yet, our SLR has revealed other seminal papers that propose practical design and evaluation criteria for SSI implementations that may be more actionable. Our interviews with practitioners, who work on the adoption of SSI in the public and private sector, allowed us to incorporate their experiences into our assessment.

Using the lens provided by the MLP, a key insight from our iterative DSR evaluation was that different types of regimes apply selection criteria at different velocities. Instead of continuously stabilizing the outcome of successive learning processes to turn innovation into a new regime, the policy regime forced a breakthrough in the implementation of SSI by taking advantage of a perceived "window of opportunity" (Geels, 2004; Geels and Schot, 2007). In the meantime, both the socio-cultural regime and technological regime are still at the stage of negotiation, not yet having produced a dominant design (Sedlmeir et al., 2021; Weigl et al., 2022). This was reflected in our interviews, where several interviewees emphasized that their recommendation on how to best implement SSI-based digital identity management solutions relies on their learning from ongoing IT-projects. Specifically, this involved integration into legacy identity and access management solutions and regulatory constraints. Knowing that SSI is still in a trial phase, and that its long-term success is dependent on negotiation with selection factors of the incumbent socio-technical regime, the interviewees appreciated the overall structure of our nine DPs. Yet, they also indicated that the definitions may require adaption over time as this space becomes increasingly mature.

Our study thus contributes to various levels of the current research discussions. Theoretically, it presents a novel way of combining a constructivist theoretical lens from STS with the design science paradigm. Thereby, it adds to the epistemological diversity in the Information Systems field. As a result, our study does not only address the gap of a missing theory or framework on identity management, it also introduces a new theoretical perspective of kernel theory development. It does this through critical reflection about the materiality and non-materiality of the observed construct, thus bypassing the positivist and techno-centric presumptions that often form the basis of DSR (McKay and Marshall, 2005; Niehaves, 2007). Practical implications, on the other hand, can be drawn from the iterative refinement of our DPs with the interview partners. They provide a common denominator for research on SSI and the development and evaluation of corresponding identity management systems in practice. The final DPs also allow us to identify several tensions that may be relevant for both researchers and practitioners. These tensions not only pertain to the novelty of SSI but also to the selection environment created by the incumbent regime and the larger exogenous socio-technical landscape of the MLP (Geels, 2004; Geels and Schot, 2007). The tensions also reflect and align with the findings of Weigl et al. (2022), who studied the interpretive flexibility of SSI. Hence, we believe that these tensions represent promising research directions.

Firstly, we observed a tension between selection factors of the policy regime and the socio-cultural regime. The establishment of Data Privacy (DP5) and User Control (DP2) in SSI-based digital identity management solutions may compromise its Applicability (DP6, DP7): For example, aspects such as the theft or sharing of mobile devices were often not sufficiently considered by the originators of this concept. These originators tended to be libertarians and cryptographers whose focus was often on ensuring control and in particular minimal disclosure and anonymity. The result was a lack of unique identifiers for processes that organizations need to consider in practical applications (Allen, 2016; Camenisch and Lysyanskaya, 2001; Cameron, 2005). To mitigate the risk of identity-related fraud with stolen mobile devices or credentials, Tobin (2017) and Koens and Meijer (2018) suggest revocation and escrow mechanisms if credentials are used in an unlawful way or if they contradict the user-specific consistency of credentials (Camenisch and Lysyanskaya, 2001). To retain a high level of privacy, zero-knowledge proofs enable minimum disclosure while compliant with regulation that requires the verification and authentication of a certain amount of user data (Sedlmeir et al., 2021). Yet, the tools currently available for zero-knowledge proofs are difficult to integrate into existing secure elements that facilitate hardware-binding (Schellinger et al., 2022). This currently still leads to a trade-off between privacy and authenticity that – despite the availability of technical solutions (Delignat-Lavaud et al., 2016; Rosenberg et al., 2023) – has not yet been resolved in practical implementations.

A second tension arises from the conflicting selection forces of the policy regime and the socio-cultural regime. The challenge pertains to the requirement to balance Verifiability (DP6) and Reliability (DP8) against end-user expectations like Control (DP2) and Privacy (DP5). This tension has its roots in the libertarian ideals of minimal disclosure, anonymity support, and full control of users over displayed data – ideals that are commonly associated with SSI (Allen, 2016; Preukschat and Reed, 2021; Weigl et al., 2022). While a milder version of these ideals forms the core of SSI, the verifiable credentials stored in the users' wallets require a trustworthy issuer and a proof of this originator. Trust registries and qualified electronic signatures, as, for instance, implemented in the context of eIDAS, may mediate this tension in the practical implementation of SSI (Schwalm et al., 2022). Should an organization issue an incorrect attestation – whether intentionally or not – the option for revocation must be available (Interviewee 10). It should also be possible to remove an unreliable issuer from certain trust registries. As a result, abandoning information silos is only practical in the cross-domain sense: While issuers are no more involved in verifiable presentations, they still need to store some of the attestation-related information to facilitate potential future revocation.

A third tension emerges from selection factors of the socio-cultural and the technological regimes. This tension pertains to the balance between the desire for maximum flexibility and the functional requirements of Interoperability (DP3). With an initially strong focus on libertarian values (Allen, 2016), the conceptual version of SSI emphasized a high degree of freedom and personalization of the technological application for users (Preukschat and Reed, 2021). This, however, makes interoperability between solutions cumbersome and impairs the desired flexibility to choose a solution that fits individual needs. Consequently, one currently "cannot copy credentials from wallet to wallet [. . .] and if you want to switch your identity to a different network, that requires reissuing the credentials on the other network" (Interviewee 10). A more "mainstream" version of SSI, thus, would have to mediate between flexibility and interoperability by enforcing some degree of standardization, yet without hampering the portability of digital wallets that hold the cryptographic keys and credentials to avoid vendor lock-in (Allen, 2016; Ferdous et al., 2019; Koens and Meijer, 2018; Yan et al., 2017).

Our DSR study contextualizes the current development and discusses factors that helped develop SSI as a new regime of identity management from a broad, transnational perspective. Yet, we cannot guarantee that we incorporated all relevant events and practical implementations of SSI in this study. We aimed to ensure a comprehensive perspective via using broad search strings, many databases, and forward and backwards searches in our SLR. During the interviews that guided the refinement of DPs, we made inquiries about other interviewees or projects that may be of relevance. Nevertheless, it should be noted that, with the exception of one Asian researcher, all our interview partners were European and North American. Moreover, the interviews were distributed only over 6 months. A more longitudinal study that rigorously analyzes discussions from events (such as the latest IIWs) or amendments in regulatory documents) may be required to consolidate the chronology of changes. Our DPs form a snapshot of the current design knowledge on SSI and a perspective on its pathway through regimes of identity management. Yet, they may be subject to change, not least, from advances in knowledge gained from successful or failed applications of SSI. We will seek better retracing of the selection factors of each regime by conducting further interviews with experts in the respective regimes. In addition, to grasp the considerations of the socio-cultural regime and that of end-users, future research may add a survey-based evaluation.

H.8 Conclusion

Our study retraces the historical development of SSI using the MLP as a theoretical lens. Our SLR in combination with DSR delivered a set of nine DPs that consolidate existing design knowledge of the SSI concept. We refined and extended this consolidated knowledge in four iterations with 15 experts from industry and academia. We used the MLP as a frame to help us to better understand the development of the concept of SSI. It was originally introduced mainly by a radical niche, but is now widely taken into account by states and industy consortia. Use currently seems focused in North America and Europe, including the eIDAS 2.0 regulation designed for large-scale productive use. Our work may help to better understand SSI in the context of business and regulated domains and to communicate its key characteristics and technical building blocks to decision makers and end-users. We also discovered tensions between the different negotiating regimes and suggested ways to mediate these. In this context, we elaborated on the difficulties that different velocities of regime negotiation could have on the prudent use of windows of opportunity. The relevance of our research comes from the close interaction with stakeholders who take part in projects in the SSI ecosystem. Aside from direct experience, our research also draws on observations from crucial requirements and real-life failures, as illustrated, for instance, by the German government's digital driver's license. While the knowledge gained from this, and changes to the concept may initially seem to considerably impair SSI's key goal of giving users more control, it also contributed to establishing an open ecosystem of verifiable digital interaction. We learned that if SSI aims to embrace digital identity management in practice, updates to its core principles are indispensable. By establishing consensus on an updated model of SSI that is integrated in regulatory and institutional requirements, our findings also suggest that a perception of SSI as a concept driven by anti-democratic forces owing to its name may be a minor issue (Sedlmeir et al., 2021). Consequently, our contribution indicates that research that consolidates historical influences on SSI may help to mediate tensions and contribute to achieving a feasible identity management solution beyond authentication (Bonneau et al., 2012). Our DPs also aim to provide a common basis for future research on design choices and trends within decentralized digital identity systems. Based on such a common understanding, researchers may tackle some of the remaining open questions concerning the design of SSI-based solutions. This involves, among others, further studying user experience requirements and corresponding success factors (Sartor et al., 2022), investigating the necessity of improved anonymous credential implementations with extended privacy capabilities (Rosenberg et al., 2023), and studying the fitness of technical tools like blockchain for decentralized governance, enhanced availability, or social recovery (Benchaya Gans et al., 2022).

Acknowledgements

We thank our interview partners for their time and dedication and their valuable insights. Further, we want to express our gratitude to Alexander Rieger and the anonymous reviewers for their constructive feedback. This work was supported by PayPal and the Luxembourg National Research Fund FNR (P17/IS/13342933/PayPal-FNR/Chair in DFS/Gilbert Fridgen). Moreover, we would like to acknowledge the support of the European Union (EU) within its Horizon 2020 programme, project MDOT (Medical Device Obligations Taskforce), Grant agreement 814654.

References

Abdullah, A., S. d. Breeijen, K. Cooper, M. Corning, O. Coutts, R. Cranston, H. Dahl, D. Hardman, N. Hickman, and N. Neubauer (2019). *On guardianship in self-sovereign identity*. URL: https://sovrin.org/wp-content/uploads/Guardianship-Whitepaper.pdf.

Abraham, A. (2017). *Whitepaper about the concept of self-sovereign identity including its potential*. URL: https://www.egiz.gv.at/files/download/Self-Sovereign-Identity-Whitepaper.pdf.

Adams, C. and S. Lloyd (2003). *Understanding PKI: Concepts, standards, and deployment considerations*. Addison-Wesley Professional.

Allen, C. (2016). *The path to self-sovereign identity*. Life with Alacrity. URL: http://www.lifewithalacrity.com/2016/04/the-path-to-self-sovererign-identity.html.

Alpár, G. and B. Jacobs (2013). "Towards practical attribute-based identity management: The IRMA trajectory". In: *IFIP Working Conference on Policies and Research in Identity Management*. Springer. DOI: 10.1007/978-3-642-37282-7_1.

Alsayed Kassem, J., S. Sayeed, H. Marco-Gisbert, Z. Pervez, and K. Dahal (2019). "DNS-IdM: A blockchain identity management system to secure personal data sharing in a network". In: *Applied Sciences* 9 (15). DOI: 10.3390/app9152953.

Backes, M., J. Camenisch, and D. Sommer (2005). "Anonymous yet accountable access control". In: *Proceedings of the Workshop on Privacy in the Electronic Society*. ACM, pp. 40–46. DOI: 10.1145/1102199.1102208.

Benchaya Gans, R., J. Ubacht, and M. Janssen (2022). "Governance and societal impact of blockchain-based self-sovereign identities". In: *Policy and Society* 41 (3), pp. 402–413. DOI: 10.1093/polsoc/puac018.

Bonneau, J., C. Herley, P. C. Van Oorschot, and F. Stajano (2012). "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes". In: *Symposium on Security and Privacy*. IEEE, pp. 553–567. DOI: 10.1109/SP.2012.44.

Callon, M. (1986). "The sociology of an actor-network: The case of the electric vehicle". In: *Mapping the Dynamics of Science and Technology*. Palgrave Macmillan, pp. 19–34. DOI: 10.1007/978-1-349-07408-2_2.

Camenisch, J. and A. Lysyanskaya (2001). "An efficient system for non-transferable anonymous credentials with optional anonymity revocation". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, pp. 93–118. DOI: 10.1007/3-540-44987-6_7.

Cameron, K. (2005). *The laws of identity*. Kim Cameron's Identity Weblog. URL: https://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf.

Cavoukian, A. (2009). *Privacy by design... Take the challenge*. Information and Privacy Commissioner. URL: https://www.ipc.on.ca/wp-content/uploads/Resources/PbDReport.pdf.

Chadwick, D., A. Otenko, and E. Ball (2003). "Role-based access control with X.509 attribute certificates". In: *IEEE Internet Computing* 7 (2), pp. 62–69. DOI: 10.1109/MIC.2003.1189190.

Chaum, D. (1985). "Security without identification: Transaction systems to make Big Brother obsolete". In: *Communications of the ACM* 28 (10), pp. 1030–1044. DOI: 10.1145/4372.4373.

Clauß, S. and M. Köhntopp (2001). "Identity management and its support of multilateral security". In: *Computer Networks* 37 (2), pp. 205–219. DOI: 10.1016/S1389-1286(01)00217-1.

Collingridge, D. (1980). *The Social Control of Technology*. Open University Press.

Čučko, Š. and M. Turkanović (2021). "Decentralized and self-sovereign identity: Systematic mapping study". In: *IEEE Access* 9, pp. 139009–139027. DOI: 10.1109/access.2021.3117588.

Delignat-Lavaud, A., C. Fournet, M. Kohlweiss, and B. Parno (2016). "Cinderella: Turning shabby X.509 certificates into elegant anonymous credentials with the magic of verifiable computation". In: *Symposium on Security and Privacy*. IEEE, pp. 235–254. DOI: 10.1109/SP.2016.22.

Dijck, J. van and B. Jacobs (2020). "Electronic identity services as sociotechnical and political-economic constructs". In: *New Media & Society* 22 (5), pp. 896–914. DOI: 10.1177/146144481987253.

El Maliki, T. and J.-M. Seigneur (2007). "A survey of user-centric identity management technologies". In: *International Conference on Emerging Security Information, Systems, and Technologies*. IEEE, pp. 12–17. DOI: 10.1109/SECUREWARE.2007.4385303.

Ferdous, M. S., F. Chowdhury, and M. O. Alassafi (2019). "In search of self-sovereign identity leveraging blockchain technology". In: *IEEE Access* 7, pp. 103059–103079. DOI: 10.1109/access.2019.2931173.

Feulner, S., J. Sedlmeir, V. Schlatt, and N. Urbach (2022). "Exploring the use of self-sovereign identity for event ticketing systems". In: *Electronic Markets* 32 (3), pp. 1759–1777. DOI: 10.1007/s12525-022-00573-9.

Fink, A. (2019). *Conducting research literature reviews: From the Internet to paper*. 5th ed. SAGE.

Geels, F. W. (2002). "Technological transitions as evolutionary reconfiguration processes: A multi-level perspective and a case-study". In: *Research Policy* 31 (8-9), pp. 1257–1274. DOI: 10.1016/S0048-7333(02)00062-8.

Geels, F. W. (2004). "From sectoral systems of innovation to socio-technical systems". In: *Research Policy* 33 (6–7), pp. 897–920. DOI: 10.1016/j.respol.2004.01.015.

Geels, F. W. and J. Schot (2007). "Typology of sociotechnical transition pathways". In: *Research Policy* 36 (3), pp. 399–417. DOI: 10.1016/j.respol.2007.01.003.

Goodell, G. and T. Aste (2019). "A decentralized digital identity architecture". In: *Frontiers in Blockchain* 2. DOI: 10.3389/fbloc.2019.00017.

Gregor, S. and A. R. Hevner (2013). "Positioning and presenting design science research for maximum impact". In: *MIS Quarterly* 37 (2), pp. 337–355. DOI: 10.25300/misq/2013/37.2.01.

Grüner, A., A. Mühle, T. Gayvoronskaya, and C. Meinel (2019). "A comparative analysis of trust requirements in decentralized identity management". In: *International Conference on Advanced Information Networking and Applications*. Springer, pp. 200–213. DOI: 10.1007/978-3-030-15032-7_18.

Hardman, D. (2019). *What if someone steals my phone?* URL: https://sovrin.org/wp-content/uploads/2019/03/What-if-someone-steals-my-phone-110319.pdf.

Hevner, A., S. T. March, J. Park, S. Ram, et al. (2004). "Design science research in information systems". In: *MIS Quarterly* 28 (1), pp. 75–105. DOI: 10.2307/25148625.

Hughes, T. P. (1983). *Networks of Power: Electrification in Western Society, 1880-1930*. John Hopkins University Press.

Jørgensen, K. P. and R. Beck (2022). "Universal wallets". In: *Business & Information Systems Engineering* 64, pp. 115–125. DOI: 10.1007/s12599-021-00736-6.

Jøsang, A. (2014). "Identity management and trusted interaction in Internet and mobile computing". In: *IET Information Security* 8 (2), pp. 67–79. DOI: 10.1049/iet-ifs.2012.0133.

Koens, T. and S. Meijer (2018). *Matching identity management solutions to self-sovereign identity principles*.

Kubach, M., C. H. Schunck, R. Sellung, and H. Roßnagel (2020). "Self-sovereign and decentralized identity as the future of identity management?" In: *Open Identity Summit 2020*. Gesellschaft für Informatik eV. DOI: 10.18420/ois2020_03.

Kuechler, W. and V. Vaishnavi (2012). “A framework for theory development in design science research: Multiple perspectives”. In: *Journal of the Association for Information Systems* 13 (6), pp. 395–423. DOI: 10.17705/1jais.00300.

Kuperberg, M. (2019). “Blockchain-based identity management: A survey from the enterprise and ecosystem perspective”. In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1008–1027. DOI: 10.1109/tem.2019.2926471.

Kuperberg, M., S. Kemper, and C. Durak (2019). “Blockchain usage for government-issued electronic IDs: A survey”. In: *Proceedings of the 31st International Conference on Advanced Information Systems Engineering*. Springer, pp. 155–167. DOI: 10.1007/978-3-030-20948-3_14.

Lacity, M. and E. Carmel (2022). *Implementing self-sovereign identity (SSI) for a digital staff passport at UK NHS*. URL: https://cpb-us-e1.wpmucdn.com/wordpressua.uark.edu/dist/5/444/files/2018/01/BCoE2022SS1FINAL.pdf.

Lacity, M. C. (2022). “Blockchain: From Bitcoin to the Internet of Value and beyond”. In: *Journal of Information Technology* 37 (4). DOI: 10.1177/02683962221086300.

Madsen, P., Y. Koga, and K. Takahashi (2005). “Federated identity management for protecting users from ID theft”. In: *Proceedings of the Workshop on Digital Identity Management*, pp. 77–83. DOI: 10.1145/1102486.1102500.

Maler, E. and D. Reed (2008). “The Venn of identity: Options and issues in federated identity management”. In: *IEEE Security & Privacy* 6 (2), pp. 16–23. DOI: 10.1109/msp.2008.50.

McKay, J. and P. Marshall (2005). “A review of design science in information systems”. In: *Proceedings of the 16th Australasian Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/acis2005/5.

Miles, M. B., A. M. Huberman, and J. Saldaña (2018). *Qualitative data analysis: A methods sourcebook*. 4th ed. SAGE.

Moe, K. S. and M. Thwe (2019). “Investigation of blockchain based identity system for privacy preserving university identity management system”. In: *International Journal of Trend in Scientific Research and Development* 3 (6), pp. 336–341. URL: https://www.ijtsrd.com/papers/ijtsrd28095.pdf.

Mühle, A., A. Grüner, T. Gayvoronskaya, and C. Meinel (2018). “A survey on essential components of a self-sovereign identity”. In: *Computer Science Review* 30, pp. 80–86. DOI: 10.1016/j.cosrev.2018.10.002.

Myers, M. D. and M. Newman (2007). “The qualitative interview in IS research: Examining the craft”. In: *Information and Organization* 17 (1), pp. 2–26. DOI: 10.1016/j.infoandorg.2006.11.001.

Narayanan, A. (2013). “What happened to the crypto dream?, Part 1”. In: *IEEE Security & Privacy* 11 (2), pp. 75–76. DOI: 10.1109/MSP.2013.45.

Niehaves, B. (2007). “On epistemological diversity in design science: New vistas for a design-oriented IS research?” In: *Proceedings of the 28th International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2007/133.

OECD (2011). *Digital identity management: Enabling innovation and trust in the Internet economy*. URL: https://www.oecd.org/sti/ieconomy/49338380.pdf.

Peffers, K., T. Tuunanen, and B. Niehaves (2018). “Design science research genres: Introduction to the special issue on exemplars and criteria for applicable design science research”. In: *European Journal of Information Systems* 27 (2), pp. 129–139. DOI: 10.1080/0960085X.2018.1458066.

Peffers, K., T. Tuunanen, M. A. Rothenberger, and S. Chatterjee (2007). “A design science research methodology for information systems research”. In: *Journal of Management Information Systems* 24 (3), pp. 45–77. DOI: 10.2753/mis0742-1222240302.

Preukschat, A. and D. Reed (2021). *Decentralized digital identity and verifiable credentials: Self-sovereign identity*. Manning.

Rannenberg, K., J. Camenisch, and A. Sabouri (2015). *Attribute-based credentials for trust – Identity in the information society*. Springer. DOI: 10.1007/978-3-319-14439-9.

Rieger, A., T. Roth, J. Sedlmeir, and G. Fridgen (2021). “The privacy challenge in the race for digital vaccination certificates”. In: *Med* 2 (6), pp. 633–634. DOI: 10.1016/j.medj.2021.04.018.

Rivest, R. L., A. Shamir, and L. Adleman (1978). “A method for obtaining digital signatures and public-key cryptosystems”. In: *Communications of the ACM* 21 (2), pp. 120–126. DOI: 10.1145/359340.359342.

Rosenberg, M., J. White, C. Garman, and I. Miers (2023). “zk-creds: Flexible anonymous credentials from zkSNARKs and existing identity infrastructure”. In: *Symposium on Security and Privacy*. IEEE. DOI: 10.1109/SP46215.2023.00108.

Sartor, S., J. Sedlmeir, A. Rieger, and T. Roth (2022). “Love at first sight? A user experience study of self-sovereign identity wallets”. In: *Proceedings of the 30th European Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/ecis2022_rp/46/.

Schellinger, B., J. Sedlmeir, L. Willburger, J. Strüker, and N. Urbach (2022). *Mythbusting Self-Sovereign Identity (SSI): Diskussionspapier zu selbstbestimmten digitalen Identitäten*. URL: https://eref.uni-bayreuth.de/68552.

Schlatt, V., J. Sedlmeir, S. Feulner, and N. Urbach (2022). “Designing a framework for digital KYC processes built on blockchain-based self-sovereign identity”. In: *Information & Management* 59 (7). DOI: 10.1016/j.im.2021.103553.

Schwalm, S., D. Albrecht, and I. Alamillo (2022). “eIDAS 2.0: Challenges, perspectives and proposals to avoid contradictions between eIDAS 2.0 and SSI”. In: *Open Identity Summit 2022*. Gesellschaft für Informatik eV, pp. 63–74. DOI: 10.18420/OID2022_05.

Sedlmeir, J., J. Lautenschlager, G. Fridgen, and N. Urbach (2022). “The transparency challenge of blockchain in organizations”. In: *Electronic Markets* 32 (3), pp. 1779–1794. DOI: 10.1007/s12525-022-00536-0.

Sedlmeir, J., R. Smethurst, A. Rieger, and G. Fridgen (2021). “Digital identities and verifiable credentials”. In: *Business & Information Systems Engineering* 63 (5), pp. 603–613. DOI: 10.1007/s12599-021-00722-y.

Seltsikas, P. and R. M. O’Keefe (2010). “Expectations and outcomes in electronic identity management: The role of trust and public value”. In: *European Journal of Information Systems* 19 (1), pp. 93–103. DOI: 10.1057/ejis.2009.51.

Smith, H. A. and J. D. McKeen (2011). “The identity management challenge”. In: *Communications of the Association for Information Systems* 28 (1), pp. 169–180. DOI: 10.17705/1CAIS.02811.

Soltani, R., U. T. Nguyen, and A. An (2018). “A new approach to client onboarding using self-sovereign identity and distributed ledger”. In: *International Conference on Internet of Things and Green Computing and Communications and Cyber, Physical and Social Computing and Smart Data*. IEEE, pp. 1129–1136. DOI: 10.1109/cybermatics_2018.2018.00205.

Soltani, R., U. T. Nguyen, and A. An (2021). “A survey of self-sovereign identity ecosystem”. In: *Security and Communication Networks*. DOI: 10.1155/2021/8873429.

Sonnenberg, C. and J. vom Brocke (2012). “Evaluations in the science of the artificial – reconsidering the build-evaluate pattern in design science research”. In: *International Conference on Design Science Research in Information Systems*. Springer, pp. 381–397. DOI: 10.1007/978-3-642-29863-9_28.

Sovrin Foundation (2021). *Principles of SSI v2*. URL: https://sovrin.org/principles-of-ssi/.

Stokkink, Q. and J. Pouwelse (2018). “Deployment of a blockchain-based self-sovereign identity”. In: *International Conference on Internet of Things and Green Computing and Communications and Cyber, Physical and Social Computing and Smart Data*. IEEE, pp. 1336–1342. DOI: 10.1109/Cybermatics_2018.2018.00230.

Tobin, A. (2017). *Sovrin: What goes on the ledger?* URL: https://sovrin.org/wp-content/uploads/2017/04/What-Goes-On-The-Ledger.pdf.

Tobin, A. and D. Reed (2016). *The inevitable rise of self-sovereign identity*. The Sovrin Foundation. URL: https://sovrin.org/wp-content/uploads/2017/06/The-Inevitable-Rise-of-Self-Sovereign-Identity.pdf.

Toth, K. C. and A. Anderson-Priddy (2019). "Self-sovereign digital identity: A paradigm shift for identity". In: *IEEE Security & Privacy* 17 (3), pp. 17–27. DOI: 10.1109/msec.2018.2888782.

Trust over IP Foundation (2021). *Principles of SSI*. URL: https://trustoverip.org/wp-content/uploads/2021/10/ToIP-Principles-of-SSI.pdf.

van Bokkem, D., R. Hageman, G. Koning, L. Nguyen, and N. Zarin (2019). *Self-sovereign identity solutions: The necessity of blockchain technology*. URL: https://arxiv.org/abs/1904.12816.

vom Brocke, J., R. Winter, A. Hevner, and A. Maedche (2020). "Special issue editorial – Accumulation and evolution of design knowledge in design science research: A journey through time and space". In: *Journal of the Association for Information Systems* 21 (3), pp. 520–544. DOI: 10.17705/1jais.00611.

Webster, J. and R. T. Watson (2002). "Analyzing the past to prepare for the future: Writing a literature review". In: *MIS Quarterly* 26 (2), pp. 13–26. URL: https://www.jstor.org/stable/4132319.

Weigl, L., T. J. Barbereau, A. Rieger, and G. Fridgen (2022). "The social construction of self-sovereign identity: An extended model of interpretive flexibility". In: *Proceedings of the 55th Hawaii International Conference on System Sciences*, pp. 2543–2552. DOI: 10.24251/hicss.2022.316.

Whitley, E. A. (2009). "Informational privacy, consent and the 'control' of personal data". In: *Information Security Technical Report* 14 (3), pp. 154–159. DOI: 10.1016/j.istr.2009.10.001.

Whitley, E. A., U. Gal, and A. Kjaergaard (2014). "Who do you think you are? A review of the complex interplay between information systems, identification and identity". In: *European Journal of Information Systems* 23 (1), pp. 17–35. DOI: 10.1057/ejis.2013.34.

Windley, P. J. (2019). "Multisource digital identity". In: *IEEE Internet Computing* 23 (5), pp. 8–17. DOI: 10.1109/MIC.2019.2940222.

Yan, Z., G. Gan, and K. Riad (2017). "BC-PDS: Protecting privacy and self-sovereignty through blockchains for OpenPDS". In: *Symposium on Service-Oriented System Engineering*. IEEE, pp. 138–144. DOI: 10.1109/SOSE.2017.30.

Zimmermann, P. R. (1995). *The official PGP user's guide*. MIT Press.

Zuboff, S. (2015). "Big other: Surveillance capitalism and the prospects of an information civilization". In: *Journal of Information Technology* 30 (1), pp. 75–89. DOI: 10.1057/jit.2015.5.

J Research Paper 8 – Designing a framework for digital KYC processes built on blockchain-based self-sovereign identity

Authors:

Vincent Schlatt, Johannes Sedlmeir, Simon Feulner, Nils Urbach

Published in:

Information & Management (DOI: 10.1016/j.im.2021.103553)

Abstract:

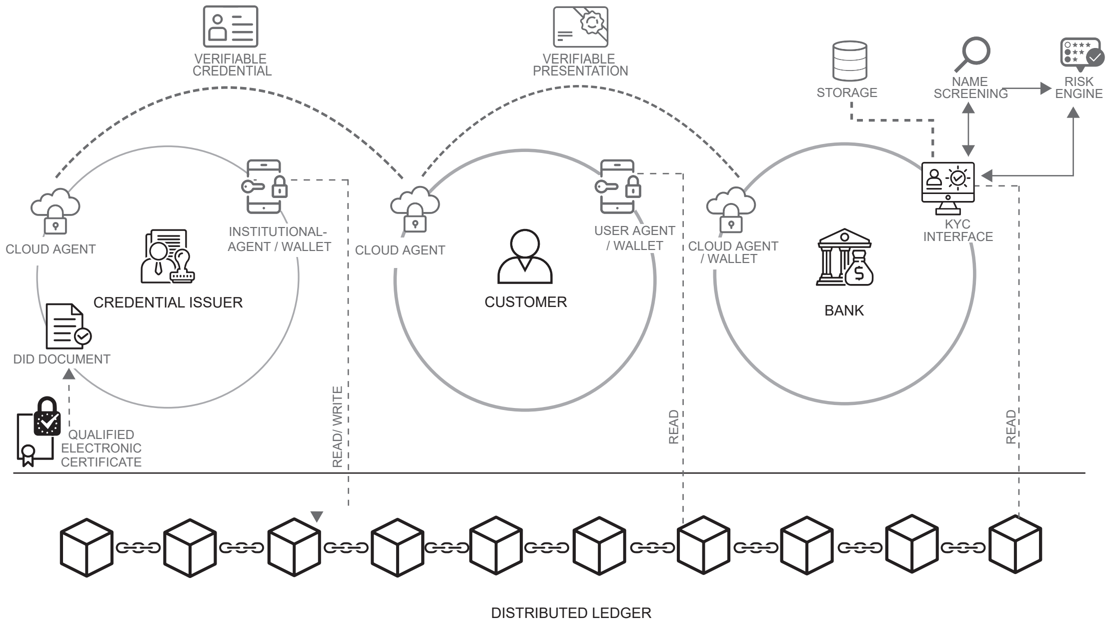
Know your customer (KYC) processes place a great burden on banks, because they are costly, inefficient, and inconvenient for customers. While blockchain technology is often mentioned as a potential solution, it is not clear how to use the technology's advantages without violating data protection regulations and customer privacy. We demonstrate how blockchain-based self-sovereign identity (SSI) can solve the challenges of KYC. We follow a rigorous design science research approach to create a framework that utilizes SSI in the KYC process, deriving nascent design principles that theorize on blockchain's role for SSI.

Keywords:

Banking, digital certificate, digital wallet, decentralized identity, distributed ledger technology, verifiable credential

Highlights

- SSI can make KYC processes completely digital, efficient, compliant, and convenient.
- No personal data need to be stored on a blockchain.
- SSI inhibits data silos, lock-in effects, and aggregation of market power.
- Blockchain's role for SSI should be more restricted than is often proposed.



J.1 Introduction

Financial regulation has three primary goals: financial inclusion, financial stability, and market integrity (Zetzsche et al., 2018). To achieve the goal of market integrity, regulators have introduced several regulatory requirements into the financial sector, such as the Financial Action Task Force on Money Laundering (FATF) recommendations, which seek to prevent money laundering and the financing of international terrorism, as well as Basel III, in reaction to the global financial crisis in 2008 (Arner et al., 2016). To remain compliant with this regulatory regime, financial institutions must perform in-depth due diligence to identify their customers and to understand the purpose of their activities, a process formally known as know your customer (KYC) (Arasa and Ottichilo, 2015), in which customers typically need to be physically present at the bank's branch or on a video call to provide personally identifying information, such as a passport or an ID card.

This process is problematic for banks, because it is cost-intensive, time-consuming, and inconvenient for customers (Zetzsche et al., 2018). Thus, there have been several attempts at improvement, mostly involving the digitization of particular process steps. For instance, some banks use their customers' analog proof of identity, such as passports, and create internally used digital customer identities to improve the process flow. However, this approach again suffers from inefficiencies, since it is error-prone, time-consuming (Jessel et al., 2018), and highly repetitive (Zetzsche et al., 2018). The lack of shared standards and banks' reservations about sharing customer information with competitors also limit the reusability of a customer's KYC data at different banks (Arner et al., 2019).

A central utility that collects and provides identity-related data for an electronic KYC (eKYC) process, as in India or Australia, is often mentioned as a solution to the aforementioned problems (Arner et al., 2019; Perlman and Gurung, 2019; Zetzsche et al., 2018), since it can reduce costs and significantly shorten KYC onboarding processes (Rajput and Gopinath, 2017). However, recent reports of leaks and misuses of personal data have lowered the confidence of both banks and customers in solutions that involve creating central data silos (Swinhoe, 2020). Moreover, there are jurisdictions in which such a centralized service run by the government is not feasible (Rieger et al., 2019). Generally, the fear that such a distinct service provider will aggregate significant market or political power impedes the establishment of a widely accepted centralized service provider (Zavolokina et al., 2020).

Thus, both researchers and practitioners have identified blockchain technology as a potential solution to the latter problems. Blockchains can provide neutral platforms for digital cross-organizational workflows (Guggenberger et al., 2020), mitigating the threat of market power aggregation. At the same time, blockchain technology enables digital trust through synchronized redundancy and therefore transparency, tamper-resistance, and enforcement of processes through smart contracts (Rossi et al., 2019). However, it is well known that blockchain technology's built-in transparency and append-only structure aggravates privacy-related problems (Rieger et al., 2019). Particularly, the European general data protection regulation (GDPR) grants individuals the *right to be forgotten*, which means that they can demand that their private data be deleted at any time as soon as the purpose for their storage has expired. As data stored on a blockchain practically cannot be erased, implementations such as Moyano and Ross's (2017) where eKYC-related information is stored transparently on-chain, are not a viable solution.

As an alternative, one could think of depositing the KYC information in a standardized way at the one and only entity involved in each of its KYC processes – the customer. These considerations lead to the concept of self-sovereign identity (SSI), which seeks to establish holistic digital identity management on the paradigm that a user controls all their data and attestations, similar to today's analog identity management via a system of plastic cards in physical wallets. Yet, SSI is still strongly linked with blockchain technology because it requires a neutral platform that provides governance, standards, and essential public information to check the validity of attestations. This goal of an interoperable digital identity management system without a distinct central authority in control makes SSI very attractive for digitizing the KYC process.

A recent pilot in the UK that investigated the opportunities of SSI-based KYC found that an SSI-based "portable identity significantly improves both consumer experience and protection, while accelerating customer onboarding and reducing KYC and compliance-related costs for financial institutions" (Ledger Insights, 2020). While research on the problem and approaches to SSI-based eKYC onboarding have recently emerged (Soltani et al., 2018), they have not covered topics such as user orientation, coverage of the entire KYC process, or platform independence. Further, Soltani et al. (2018) focused on implementing the principles of SSI without acknowledging that SSI is a tool to achieve an improved KYC process from the perspective of banks. Looking at SSI generally, in the related literature, blockchain's role in this context remains largely unclear. Thus, both research and practice need a generic and validated framework that guides the design of SSI solutions for entire eKYC processes and an overview of the resulting implications to assess the potential benefits and to learn how to leverage them. Further, we still lack generic design principles (DPs) to guide the development of SSI solutions based on blockchain technology that can also be used in other sectors (Liu et al., 2020). We seek to design a framework for an eKYC process built on blockchain-based SSI and to derive initial generic DPs. We develop and evaluate our framework in a rigorous design science research (DSR) approach, incorporating both existing theoretical knowledge and practitioners' perspectives through semi-structured expert interviews. Thus, we extend the literature on eKYC by providing a comprehensive architecture and process framework, discussing the roles of blockchain and SSI for eKYC, and producing generalizable knowledge on the design, opportunities, and challenges of blockchain-based SSI systems. The DPs we develop from our DSR suggest that blockchain's role in SSI should be more restrictive than is typically proposed in the literature in order to make systems scalable and compliant with regulations. We also guide practitioners on how to design the respective systems.

The remainder of this study is structured as follows: In Section J.2, we present background knowledge on KYC processes, blockchain technology, and SSI that is necessary to understand the work that follows. In Section J.3, we present our DSR method. In Section J.4, we derive objectives for the eKYC framework and evaluate them through expert interviews. We present the framework, including the SSI-based eKYC architecture and process, in Section J.5. Section J.6 continues with the evaluation of the framework along the derived objectives. In Section J.7, we discuss the findings, develop nascent DPs for blockchain-based SSI, and provide managerial and theoretical implications. In Section J.8, we summarize our results, identify limitations, and provide an outline for further research.

J.2 Background

J.2.1 The KYC process and centralized attempts at eKYC

After the original FATF Forty Recommendations were drawn up in 1990, they were revised in 1996 to account for the latest money laundering techniques. These recommendations for anti-money laundering (AML) have been adopted by more than 130 countries and are therefore considered to be the international standards (Ruce, 2011). A key element of these recommendations is the KYC process. Financial institutions are urged not to open anonymous accounts or accounts with obviously fictitious names. In this

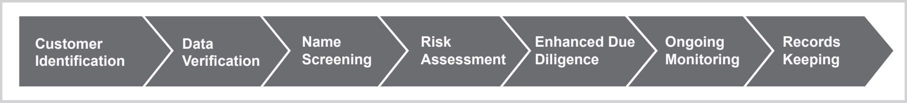


Figure 1: The KYC process.

context, due diligence is recommended to verify the identity of customers through independent, credible documents. The purpose of the business relationship must also be verified. Further, the KYC process should include ongoing monitoring of transactions to identify suspicious customer behavior (FATF, 2004).

KYC processes may differ, owing to countries' different regulatory requirements and the banks' specific requirements. However, some repeating core activities of the KYC process can be identified (see Figure 1). The process begins with the collection of data about potential customers to identify them. Government-issued documents such as ID cards, driver's licenses, or passports are preferred. Documents from other companies in the financial sector, as well as other documents relevant for the identification of persons, such as telephone or gas invoices, can also be used (Mugarura, 2014). After a customer is identified and the identity data claims are verified, the bank checks whether the person represents a risk for the financial institution. This includes matching against a list of known terrorists, criminals, and politically exposed persons (Arasa and Ottichilo, 2015). Further initial and ongoing measures follow to allow the bank to do permanent risk monitoring.

The process of initial verification and ongoing monitoring of activities must be repeated for each customer, and every customer must undergo this process again when opening an account with a new bank. Thus, the KYC process is very time-intensive and inconvenient for both customers and banks, resulting in poor customer experiences and fewer account openings. For instance, 89 % of surveyed customers did not have a good experience with the KYC process (Thomson Reuters, 2016), and criticized the onboarding process because it was time-intensive and involved posting several documents. To avoid losing their customers and revenue opportunities, financial institutions must make essential improvements. Also, the overall market efficiency could benefit from enhanced competition owing to lower switching costs. Further, the high effort required for this process and the lack of automation of some manual steps result in high costs for the financial institutions. A survey of 800 financial institutions found that the annual cost for KYC per bank is approximately USD $60 million (Thomson Reuters, 2016).

The primary focus and motivator of regulatory efforts toward KYC is the avoidance of money laundering through financial institutions. Failure to comply with regulatory requirements may further increase KYC process cost through considerable fines (Moyano and Ross, 2017). A major goal in KYC efforts for financial institutions is often, therefore, the avoidance of fines or loss of reputation, at preferably low ownership costs. However, some institutions also see KYC as an opportunity, since it enables them to better understand customers, identify their needs and behaviors, create customized products, and improve customer relationships, ultimately leading to higher company profits (Ruce, 2011).

The key to simultaneously reducing compliance costs, preventing regulatory penalties, and harnessing new potential lies in the digitization and automation of processes and the resulting opportunities for data processing and analysis (Lootsma, 2017). One often used approach to improve the KYC process is the digitization of analog ID documents, which typically involves some facial verification step by a combination of human and machine learning examination. One step further are approaches that seek to abandon analog documents altogether, which is why the term eKYC is often used here (Christie, 2018). A sector-wide eKYC utility could avoid the repeated execution of the KYC process at different banks. These systems typically use biometrics such as fingerprints, iris scans, or facial recognition. The data are then stored on a smart ID card and online in a central database, together with personally identifiable information such as the customer's name, age, and place of residence. During the KYC process, the customer's biometric data are captured and matched against the data in the central online database.

An example of such a sector-wide eKYC utility is India's Aadhaar system. Indian citizens must provide various demographic and biometric data (Zetzsche et al., 2018), which are stored in a central database. The system has led to much faster onboarding times and to fewer losses from fraud and corruption (The Economist, 2016). However, several data breaches have raised questions regarding the privacy and security of the system (Zetzsche et al., 2018). Further, if operated by a public authority or heavily regulated, identity systems such as India's could be used by governments for mass surveillance without citizens' knowledge. On the other hand, on an international level or if operated by a private company, threats of monopolies and market power may keep banks from participating in such a system (Zavolokina et al., 2020). Thus, while it is a key step toward increasing process efficiency, the design of identity systems is critical for their success and acceptance, because this involves the management of highly sensitive data and misuse should be prevented. Centralized databases to store users' personal data are attractive targets for attackers who can steal large amounts of sensitive information. Consequently, they are challenging to secure (Sedlmeir et al., 2021b). Similar concerns regarding the creation of centralized service providers for non-competitive data arise in various further contexts beyond KYC. For the KYC procedure, digital identity systems must therefore be considered from the perspectives not only of efficiency and user experience but also privacy and security (Perlman and Gurung, 2019).

J.2.2 Blockchain technology and decentralized approaches to eKYC

Owing to the limitations and downsides of centralized platforms, banks have started looking for alternatives, one being distributed ledger technology (DLT). Key components of DLT are a peer-to-peer network where all data are replicated across multiple peers, and an associated consensus protocol operated by specific nodes to ensure the validity of state modifications (*transactions*) and to synchronize all replicas (Glaser, 2017; Kolb et al., 2020). Authentication on DLT is conducted through public key cryptography, which allows one to participate in consensus or to interact with the network and authorize transactions. Distributed ledgers are resistant to crashes and even the malicious behavior of a small subset of nodes, making them a highly available and decentralized digital infrastructure. However, DLT also has considerable drawbacks concerning scalability and privacy, owing to the redundant operation of all transactions (Kolb et al., 2020). Blockchains[1] are a special case of DLT, and are probably the most widely used. The key characteristic of blockchain architectures is that transactions are batched into blocks, and each block of data contains the previous block's hash value. The blocks therefore form an append-only structure (*chain*) with the aim of establishing a tamper-resistant historical record (Butijn et al., 2020).

Thus, blockchains can serve as a physically decentralized yet logically centralized source of truth for information, making them suitable for decentralized asset management (Rossi et al., 2019). Guaranteeing transparency and the enforcement of rules while ensuring the independence from a distinct node can be major advantages of blockchain solutions for cross-organizational workflow management (Fridgen et al., 2018). Businesses and public authorities have realized DLT's potential for the digitization of their cross-organizational processes, leading to a large number of projects (Casino et al., 2019). Considering the aforementioned generic properties of DLT, a blockchain-based neutral platform on which banks could collaborate on eKYC seemed very appealing, since this approach can eliminate the threat of monopolies. However, it aggravates privacy-related problems, since tamper resistance and redundancy imply not only that stored on-chain data are visible

[1] We use the terms blockchain and DLT interchangeably in this work.

to all nodes but also that it is practically impossible to delete on-chain data (Kolb et al., 2020; Rieger et al., 2019). It therefore does not make sense to store personal data on the ledger (Dunphy and Petitcolas, 2018) and doing so contradicts regulation such as the GDPR, which includes the *right to be forgotten*.

This fact significantly complicates the conceptual integration of a DLT into the KYC process. Biryukov et al. (2018), Moyano and Ross (2017), and Norvill et al. (2019), for instance, proposed writing a proof about the successful completion of the KYC process in the form of a hash value on a blockchain. In this concept, the de facto data are still stored in a centralized database operated by banks or a service provider. Once a bank customer has completed the KYC process, it will be sufficient for the customer to prove their identity using the hash value in the ledger. Although the efficiency of the process can thus be increased, central parties with full control of and access to the data are still necessary with these approaches, again causing the described security and privacy challenges. Further, challenges regarding the binding of cryptographic keys to customers as well as the management of permissions for exchanging customer data remain, while the benefit of using a blockchain is not yet clear, as a public key infrastructure and certificates based on digital signatures can provide tamper-proof evidence of a completed KYC process. Ostern and Riedel (2020) acknowledged the challenges of storing customer data on a blockchain in their development of a blockchain-based system for KYC to satisfy the requirements of initial coin offerings. Thereby, only the statuses of completed KYC processes are stored on a blockchain. However, in their design, the customers' identity data remain with a centralized provider specialized in KYC, and the protocol for exchanging data between the banks and the eKYC provider remains unspecified.

J.2.3 SSI and its proposed application to eKYC

Today, identification and authentication are usually carried out against a service provider using a username and password. The reason for the widespread use of this so-called *centralized* identity model lies in its simple implementation and in the full control of the service providers, who can minimize risks if no third party is involved for authentication. Users also benefit from the fact that they only have to pass on the information necessary for the context in question (Clauß and Köhntopp, 2001). However, the increasing use of internet services has made this system inconvenient for users, since they have to remember the login data for each additional service, and manual input or repeated verification processes of attributes are necessary (El Maliki and Seigneur, 2007). This leads to poor user experiences and security issues, as users tend to reuse passwords across many services. Moreover, service providers need to rely on the validity of the data provided by the customer, which can result in bad data quality and costs for fraud that cannot be traced back to a natural person. Service providers also usually store the data in large data silos – a popular target for hackers (Rajput and Gopinath, 2017).

In an attempt to improve user experience, the so-called *federated* identity model was developed (Maler and Reed, 2008). This concept allows for the use of digital identities for authentication and proof of attributes across organizational and system boundaries. An identity provider, such as Facebook or Google, manages users' digital identities and makes them available to relying parties. The fundamental prerequisite for this identity model is the establishment of a trust relationship between the identity provider and the relying party. Federated identity management improves user experience, since the users no longer have to remember a large number of user names and passwords, and only need a single sign-on (Lim et al., 2018). However, from the perspective of privacy and security, such services are even more problematic than centralized systems (Maler and Reed, 2008).

If privacy and security need to be improved, there must no longer be any central parties that have access to users' full digital identities and the associated data. Rather, control must be decentralized. By using public key cryptography, users can create their own identifiers – also known as decentralized identifiers (DIDs) – and prove control over them. Users can then append information to these identifiers. For contexts in which some attested attributes require confirmation, users can collect credentials from trusted authorities, such as government agencies, companies, or universities (Sporny et al., 2019). DIDs and the associated cryptographic keys, as well as credentials, are stored by users in so-called digital wallets, for instance on smartphones, computers, or in the cloud with a provider of their choice. Such a system is comparable to the physical credentials, e.g., plastic cards, we carry in our physical wallets (Avellaneda et al., 2019). Since users fully control their data, this approach has been called *self-sovereign* (Allen, 2016).

Such an approach requires open-source and open-standard technology (Wagner et al., 2018). Various implementations of SSI are possible and have been realized, but currently many commonly used implementations build on the DID standard being developed by the World Wide Web Consortium (W3C) (Reed et al., 2021). A DID is always associated with a DID document that contains information such as public key material used to delegate and prove ownership and control of a DID (Reed et al., 2021), and to establish a secure (encrypted) communication channel with this DID. Besides the purpose of stan-

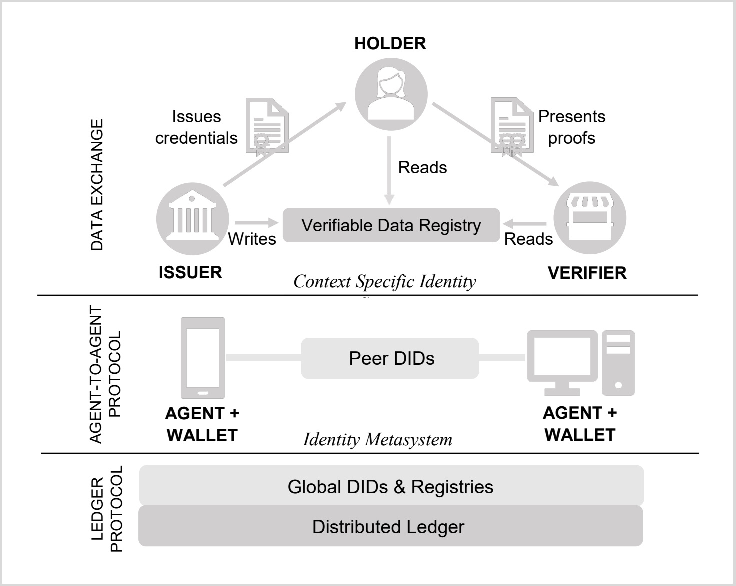


Figure 2: Layers of SSI-based identity management based on Trust over IP Foundation (2020).

dardization, DIDs create a reference point for bilateral interactions that is portable across domains and does not require a centralized authority to register, resolve, update, or revoke the identifiers (Soltani et al., 2018). In this sense, DIDs are not strictly necessary for SSI, but provide functionalities that go beyond the mere capabilities of decentralized public key infrastructure (DPKI).

Credentials that provide cryptographic evidence of who created them and who they were created for are widely known as digital certificates. A new flavor, called verifiable credentials (VCs), is currently the subject of standardization efforts by the W3C (Sporny et al., 2019). Their validity and whether they have expired or been revoked can be verified without having to communicate with the issuer of the credentials, by checking the issuer's digital signature and a public yet privacy-preserving revocation registry. However, this approach requires an established trust relationship between a verifier and the credential issuer (Mühle et al., 2018). The decentralized approach regarding the reliable and trustworthy provision of public information that is necessary to verify VC data is enabled by the use of DLT. DLT acts as a *single point of truth* and thus as a generally acceptable and immutable location for the storage and management of information about standards, issuers of VCs (e.g., their public signing keys), and revocation status. DLT therefore provides a censorship-resistant storage facility for information that must be publicly available, without the need for a central entity such as a certificate authority (Mühle et al., 2018). SSI's key roles and building blocks are summarized in Figure 2.

Besides the security aspect, a widely acknowledged opportunity of SSI is enhanced privacy features (Sedlmeir et al., 2021b). One the one hand, by default, different identifiers, so-called pairwise DIDs (pseudonyms), can be used in different interactions. Global DIDs are required only for public entities that want to aggregate reputation or trust, such as credential issuers. Further, some implementations of VCs can prove the correctness of claims, such as the existence of the issuers' signature on the VC, without the need to reveal the value of the signature itself or all attributes that are attested on the credential. This significantly mitigates the correlatability of conventional digital certificates by means of their digital signature, and ultimately allows for enhanced privacy while still exchanging the information that is required to build the trust relationship that is necessary for interactions and business (Davie et al., 2019; Hardman, 2020).

Since the concept places users in the center and leaves them in full control, some general challenges arise with SSI. First, appropriate measures must be taken to ensure user friendliness. Users must take care of storing the credentials and managing the keys themselves. So-called digital agents or wallets are used for this, either directly on an edge agent (e.g., a smartphone or laptop) or with cloud agents that can only be accessed by the user (Lyons et al., 2019). Cloud agents are helpful, since edge agents cannot guarantee permanent online availability (Reed et al., 2018). Further, problems such as recovery in the case of device loss or theft must be addressed. Second, a governance framework is required to establish the possibility to gain trust in a large variety of issuers. Third, user authenticity must be guaranteed; i.e., sharing and selling credentials must be prevented (Camenisch and Lysyanskaya, 2001). However, various concepts that allow one to address these issues, such as initiatives to build governance frameworks (Davie et al., 2019) and the combination of biometrics, cryptography, economic incentives, and device coupling to create a strong bond between credentials and holders (Hardman, 2020; Hardman et al., 2019).

In sum, SSI allows for highly decentralized management of personal identifiers and for credentials that are reusable across different contexts to be managed by the user from a single app (Sedlmeir et al., 2021b). In what follows, we investigate how this approach may help meet the challenges of the eKYC process.

J.3 Method

We followed a DSR approach. DSR was originally created to enable IS practitioners to find solutions to previously unsolved problems through a continual build-and-evaluate process. Its outcomes are IT artifacts, such as constructs, models, methods, or instantiations (Hevner et al., 2004; March and Smith, 1995). While some scholars argue that the IT artifact itself already contributes to research if it is novel and useful (Baskerville et al., 2018; Gregor and Hevner, 2013), two challenges in discerning DSR's research contribution remain: First, it is hard to determine what exactly a theoretical contribution in DSR is (Gregor and Hevner, 2013). Second, it is hard to balance concrete, practical contributions to a rapidly changing technology environment and to provide a sufficient level of generalization for theory (Baskerville et al., 2018). To address these challenges, we aim to contribute both an architectural design and a collection of processes as a concrete IT artifact (Gregor and Hevner, 2013). To elevate this IT artifact for further theoretical discussion, we then derive DPs (Gregor and Hevner, 2013; Hevner et al., 2004). Thus, we aim to contribute nascent design theory in the form of operational principles (Gregor and Hevner, 2013).

For an IT artifact to offer a substantial contribution to IS research, it must address a relevant business need (Hevner et al., 2004), which can result from the persons, organizations, or technologies used in an environment. As argued in Section J.2.1, the enhancement of the KYC process represents such a business need. However, an IT artifact must also be applicable in the corresponding environment (Hevner et al., 2004). To ensure rigor in the design process, the construction of the IT artifact needs to build on existing foundations from previous IS research (vom Brocke et al., 2020). Also, existing methodologies should be used to evaluate the created artifact (Hevner et al., 2004). The KYC framework here is based on related work that aims to improve the KYC process using digital technologies, the technical and theoretical foundations of KYC, DLT, and SSI, and the requirements and expertise of practitioners in said areas.

We employ the frequently used and widely accepted (Reinecke and Bernstein, 2013; Schweizer et al., 2017) DSR process model of Peffers et al. (2007) to facilitate the development of a relevant IT artifact created by a rigorous method. Our process has six steps arranged in sequential order (see Figure 3) and incorporates an iterative research procedure by design (Peffers et al., 2007). The process typically starts with the identification of a research problem with practical relevance. Indeed, as illustrated in Section J.2.1, our examination of the current KYC process reveals challenges such as low process efficiency, security challenges, poor user experience, and data protection concerns.

Next, we defined solution objectives to address the stated challenges and to create a meaningful artifact. In line with DSR, the insights gained from the build-and-evaluate process must be generalizable and therefore applicable in more generic settings (Jones and Gregor, 2007). Also, the design artifacts should result in profound disruptions to traditional

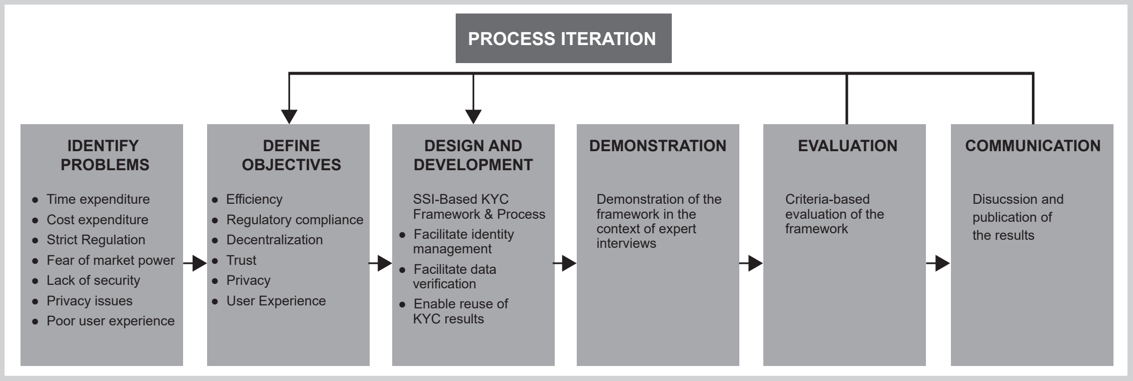


Figure 3: Our applied DSR process, following Peffers et al. (2007).

ways of doing business (Hevner and Gregor, 2020). Recent research into DSR encourages researchers to build their work on prior DSR within the respective domain (vom Brocke et al., 2020). We derived solution objectives by studying the related literature and regulatory requirements, both for the KYC process and for digital identification and authentication, resulting in six main objectives for the KYC framework and several requirements for each main objective. Based on these objectives and on theory, we design and develop an SSI-based eKYC framework in the next research process step. Phase 5 comprises evaluation, which is necessary to test whether an artifact achieves the purpose of its creation and to prove this achievement using rigorous methods (Venable et al., 2012). The evaluation phase also helps one to better understand the problem at hand and thus to realize improved outcomes (Hevner et al., 2004).

There is no unique path regarding evaluation, since the best approach depends on both the underlying problem and the artifact (Peffers et al., 2007). Our evaluation had several iterative evaluation steps, starting ex ante with the formative evaluation of the design objectives through interviews with experts (Sonnenberg and vom Brocke, 2012; Venable et al., 2016). We conducted six additional ex post interviews to summatively evaluate our framework by demonstrating it to the interviewees and incorporating their feedback. The evaluation of the framework was designed to assess its functionality, accuracy, reliability, fit with the organization, and utility (Hevner et al., 2004). We then applied a criteria-based evaluation concerning whether the derived solution objectives were met, since evaluation criteria for an IT artifact must themselves be determined for the particular environment (March and Smith, 1995). To elevate the implicit knowledge contribution in our IT artifact to more abstract and generalizable knowledge allowing for theoretical discussion (Gregor and Hevner, 2013), we then developed nascent DPs for blockchain-based SSI, as this technical approach is both novel and increasingly discussed, though

Episode	Expertise	Id	Role	Background	Type
1	KYC	A	Project Manager KYC	Strategic Analysis and Research, Banking	Phone call
1	KYC	B	Sales Director	Building Society, Banking	Video call
1	SSI	C	Identity Engineer	Innovation Consultant	Video call
2	SSI & KYC	D	Executive Director	SSI Start-up Founder, Banking	Video call
2	SSI & KYC	E	Project Manager	Banking	Phone call
2	SSI	F	Senior Developer	Computer Science	Video call
2	SSI	G	CEO	SSI Start-up Founder	Video call
2	KYC	H	Sales Executive	Banking	Video call
2	KYC	I	Sales Director	Banking	Video call

Table 1: Overview over the interviewed experts.

no general DPs currently exist in the literature. Finally, we shared the findings of our research with the relevant audience (Hevner et al., 2004). The applied DSR process was iterative and partly in parallel, since the evaluation phase's results have reshaped the created artifact (Beck et al., 2013).

Qualitative interviews, as used for our evaluation cycles, are a frequently used method in IS research, since they are suitable for generating rich data (Myers and Newman, 2007). We conducted semi-structured interviews so that we could react flexibly to the interviewees' answers and ask appropriate follow-up questions (Kallio et al., 2016)). We involved experts on KYC and SSI to reflect opinions from the perspective of practical applicability in existing settings and bank structures as well as opinions regarding technical maturity and feasibility. Also, we took care to avoid an elite bias by representing the voices of executives of different corporate levels (Myers and Newman, 2007). Further criteria for the selection of the experts included ample knowledge of their disciplines and intensive experience in their daily work, as well as the ability to provide detailed information on their field of expertise (Morse, 1991). A detailed overview over the interviewees appears in Table 1.

We recorded 320 interview minutes (an average of 35.6 minutes per interview). The interviews were recorded, transcribed, and later analyzed using MAXQDA software. For data analysis, we used both open and axial coding (Saldaña, 2015). Starting from the initial concepts originally derived in the open coding round, categories were formed. Categories are "higher-level concepts under which analysts group lower-level concepts that then become its subcategories" (Corbin and Strauss, 2008, p. 220). During this first coding round, we created 30 categories and 300 subcategories; in the second, we used axial coding to build subcategories. Thus, the data that were split up during open coding were reassembled to summarize the categories on a more abstract level (Charmaz, 2006; Corbin and Strauss, 2008; Saldaña, 2015).

J.4 Design objectives for the eKYC framework

J.4.1 Structuring of design objectives

To comprehensively address the challenges of the KYC process (as identified in Section J.2), stage 2 in our DSR process involved the derivation of objectives to be met by a useful SSI-based eKYC framework. We derived these objectives from the literature on the KYC process, KYC-related regulatory requirements, and three formative interviews with experts. Thus, we aimed to align with DSR by incorporating prior research (vom Brocke et al., 2020) and incorporating real-world business needs (Hevner et al., 2004). We identified six main objectives and associated requirements. In what follows, we explain and justify them.

Objective 1: Efficiency The high cost and human resources involved in carrying out the current KYC process strongly challenges financial institutions. Financial institutions offering fast and convenient verification of identity documents are more attractive from the customer's perspective, can reduce costs, and can help a company gain a competitive advantage (Jessel et al., 2018). To allow for increased process efficiency, we derived three requirements that had to be satisfied to overcome the existing challenges. The *end-to-end digital processing of relevant documents (R 1.1)* is a prerequisite for automating process steps and reducing friction (Arner et al., 2019). Further, in the current KYC process, many steps involving the validation of data, such as checking whether an ID document's validity has expired, are conducted manually (Zetzsche et al., 2018). Thus, the de facto *automation of manual processes (R 1.2)* is another key requirement. Further, Moyano and Ross's (2017) interviews with five senior banking executives revealed a need for interbank collaboration; this was confirmed by Experts A and B, who stated that banks would be ready to collaborate on resource-intensive KYC. Currently, however, the main barrier to such cross-organizational processes is the lack of a suitable non-proprietary IT infrastructure. Thus, a *standardized exchange of eKYC documents (R 1.3)* is crucial to allow for the efficient integration of eKYC checks that have been conducted at other institutions.

Objective 2: Regulatory compliance Compliance with regulations is a key objective of the KYC process (Ostern and Riedel, 2020); derived from the overall goal of avoiding money laundering, it is one of the main reasons why the KYC process exists at all. Our literature study revealed that the *Money Laundering Act (MLA) (R 2.1)*, *GDPR (R 2.2)*, and *electronic Identification, Authentication, and Trust Services (eIDAS) (R 2.3)* are particularly relevant regulations for a digital KYC process (Arner et al., 2019). While these apply within the European Union (EU), there are similar regulatory requirements in other jurisdictions worldwide. The European requirements are considered particularly strict, which is why we decided to apply them here.

The MLA provides banks with specific requirements regarding the identification of customers and the storage of their records. The banks are also required to determine and document the risk in relation to their customers. The GDPR applies to the processing of any data regarding natural persons, but not to legal entities, and poses requirements such as privacy by design, portability, the right to erasure, transparency, purpose limitation, data minimization, accuracy, storage limitation, information integrity, and confidentiality. Further, digital KYC processes involve the customer's identification and a check of the authenticity of the involved documents, and the 5th EU AML Directive accepts electronic ID systems that comply with eIDAS as a legitimate means of identification for KYC procedures. eIDAS imposes requirements on these electronic means of identification, such as compliance with certain security levels (level of assurance) and the cross-border interoperability of systems.

Objective 3: Decentralization As argued in Section J.2.1, silos of customer data are an attractive target for attackers. Securing valuable information is costly and not the core business of banks, and mistakes can have severe consequences concerning reputation, fines, or both. Recent data breaches that revealed sensitive customer data stored in central data silos have significantly reduced confidence in their respective architectures (Rajput and Gopinath, 2017). To avoid comparable data breaches, a viable solution for an improved eKYC process must therefore *avoid central storage of customer data (R 3.1)*. Further, banks do not want to risk becoming dependent on a centralized eKYC service provider. Thus, the system must be constructed to *prevent lock-in effects (R 3.2)* that could result in the aggregation of market power. Decentralization of both data storage and workflows is therefore one key objective of the new eKYC architecture.

Objective 4: Trust A key goal of banks is to make eKYC documents reusable in registrations of a customer at different banks. If banks do not comply with the regulations, there can be heavy fines, so it is important to establish trust in the KYC process and the integrity of its documentation at other banks. Thus, *acceptance of KYC documents attested by other banks (R 4.1)* is required. The documents must be tamper-proof, so a further requirement is that *validity checks (R 4.2)* of these documents are feasible. Another often disregarded requirement for a complete trust chain is that sharing or selling KYC documents among customers must be prevented. This can be particularly difficult if the eKYC process happens remotely and lacks interaction with an employee of the bank. The customer needs to be able to convince the bank that the KYC-related documents that they present were not stolen, sold, or shared. We call this requirement *authenticity checks (R 4.3)*, meaning that the identity of the customer and their connection with the documents must have a high level of assurance even if the customer is not present at a branch and no video call is held.

Objective 5: Privacy Protecting customers' privacy is a key feature of an eKYC process. Facing an increasing number of data leaks, customers are aware of privacy issues, and delivering a privacy preserving solution may increase the solution's acceptance. An essential and fairly universal principle in this context is *compliance with the "need to know" principle (R 5.1)*: Only the customers themselves and entities relevant to the KYC process must have access to customers' personal data. This is also a general recommendation for information systems from a security perspective (Hughes, 1988; Moor, 1997). Further, not only the parties involved in the KYC process but also the de facto data that are exchanged should be restricted to what is necessary, because digital data are much more comprehensive and easier to collect and abuse than their analog counterparts (Arner et al., 2019). We call this requirement *data minimization (R 5.2).*

Objective 6: User experience From the users' perspective, although privacy is a nice feature that can be used for marketing purposes, the most important objective is seamless user experience (Kokolakis, 2017). The eKYC process must be convenient, so that customers are not discouraged from registering at the new bank. It is only when the eKYC process is fast and simple for the customer that it can provide high security and acceptance (Dhamija and Dusseault, 2008). Thus, we made *low complexity (R 6.1)* a major requirement for user experience. Further, the variety of devices on which a customer can perform the eKYC process must be respected. Mobile phones are often the customers' preferred option, but support for web apps is also necessary in many circumstances. Thus, the availability of *different user interfaces (R 6.2)* is important. The user experience should also include exception handling, for instance, if a device that stores the customer data is lost or stolen. In this case, either there must be a built-in recovery mechanism, or the customer must be able to ask for rapid support. This is very difficult if no central third party is responsible for the whole process. Thus, we also added such *backup, recovery, and support (R 6.3)* features to our requirements.

J.4.2 Evaluation of the design objectives

We discussed the current problems of the KYC procedure and our derived objectives with two KYC experts and an SSI expert. The interviews sought to evaluate the identified design objectives concerning relevance and completeness. The KYC experts worked in different companies and held different positions, so that the objectives could be viewed from different perspectives. Additional information on the interviewees appears in Table 1.

Expert A confirmed the relevance of the derived objectives and their associated requirements. Owing to the increasing expenditure on personnel and technology, the process's efficiency is indeed a crucial goal for banks. He stressed the importance of end-to-end digital processing and advocated interbank cooperation in the KYC process, but identified trust problems here, both between the banks and concerning customer trust in the confidentiality of their data. According to him, the protection of customer privacy is also crucial. Further, he affirmed the relevance of increasingly strict regulations and the need to comply with them. For instance, customer data must be stored by banks for at least five years. The expert also confirmed the necessity of including further MLA requirements.

Expert B also described process efficiency as the most crucial factor, to ensure cost and time savings. The challenges apparently lie particularly in the high number of manual process steps. This expert emphasized the importance of automation and digital processing of documents. He also confirmed the importance of protecting privacy. Sensitive handling of customer data is necessary, and this must not be passed on to third parties, not even to cooperation partners. Like Expert A, he noted the increasing importance of regulation and the need to comply with it.

Expert C emphasized the importance of a good user experience, since many users will not focus on the systems' functional details. During the implementation phase, special care should be taken to ensure that the system is as intuitive as possible. Asked about the architectural perspective, he mentioned backup and recovery capabilities through cloud storage as a building block for user friendliness in case of data theft or loss. Expert C also confirmed the importance of the GDPR and eIDAS. According to him, there is still room for interpretation in the GDPR, for instance regarding the role of encrypted or hashed personally identifying data. He advised proceeding from the strictest possible interpretation of the GDPR. He stressed that, on a distributed ledger, data cannot be deleted. A key challenge to the acceptance of KYC documents attested by other banks, he spoke of the necessary establishment of a trust relationship between the banks. However, he argued that connecting the eKYC architecture to the eIDAS infrastructure could be a solution to this problem.

In sum, at least one expert emphasized each of the design objectives, and the experts generally considered the associated requirements useful to evaluate an eKYC framework from a bank's perspective.

J.5 A framework for eKYC processes built on blockchain-based SSI

J.5.1 The SSI-based eKYC architecture

Based on the related work presented in Section J.2, we designed a decentralized architecture that seeks to address the challenges of the KYC process. The study by (Moyano and Ross, 2017) motivated a decentralized design of eKYC to allow for inter-bank collaboration. However, the proposed system seems critical from a data protection perspective, considering the privacy-related challenges of storing customer data, also in encrypted or hashed form, on a blockchain system. We also noticed that the mechanism they presented does not enforce the alignment of incentives, since the integrity check only requires a local read operation on one node. Thus, while we appreciate the background they gave on the necessity of a reusable KYC and a non-centralized solution, we found the design of the SSI-based framework proposed by (Soltani et al., 2018) more appropriate. Nonetheless, we generalized this solution by considering both the initial onboarding to receive the first KYC document and how an existing SSI ecosystem and regulation such as eIDAS integrates with and further strengthens eKYC. We also added an investigation of the framework's practical feasibility by rigorously evaluating our SSI-based framework concerning the technical, economic, and legal requirements. From a technical perspective, we abstracted from their solution based on Hyperledger Indy to a more generic perspective on SSI and extended their findings by rigorously evaluating the design.

Following (Soltani et al., 2018) and the general approach of blockchain-based SSI, the proposed eKYC architecture involves three primary parties: the customer (holder), a bank (verifier), and an issuer (the same bank, another bank, or any third party trusted by the verifying bank, such as a government agency). Credentials from different issuers can also be conjugated, because in a so-called verifiable presentation (VP), attributes attested in different VCs can be combined (Sporny et al., 2019). For simplicity, we assumed one issuer. The customer is the KYC subject and defines the center of the architecture (see Figure 4). Customers manage their digital identity through user agents by creating and storing DIDs and cryptographic keys in their digital wallets, collecting credentials, creating backups, and managing permissions. It is possible to interact with agents on various devices, such as smartphones or laptops. At all times, the customers have full control over their data and particularly over KYC-related documents, represented by VCs. While traditional certificate-based approaches (e.g., X.509 certificates) need to be shown fully to the verifier in order to check the signature's validity, the VC standard (Sporny et al., 2019) and related implementations such as Hyperledger Aries allow for creating proofs from the VCs, convincing the verifier that certain claims extracted from the VC are correct without the need to exchange the full VC. This builds on cryptographic constructions such as anonymous credentials introduced by (Camenisch and Lysyanskaya, 2001). In our case, the VPs contain proofs of the validity of the attributes that need to be revealed during the KYC process.

To facilitate the redundant storage of credentials and easier user access to the SSI documents, as well as to enable secure communication with other entities, the framework employs cloud agents and wallets. The permissions for carrying out identification activities differ between edge and cloud instances. While edge agents and wallets are usually granted full access to an individual's data, the user should use cloud agents/wallets primarily for redundant storage and communication with other entities. A blockchain serves as a neutral infrastructure for storing publicly verifiable information. It is used to hold VC issuers' public signing keys and other institutional information. Further, schemas of KYC VCs are stored on-chain to allow for public verification. Also, publicly available revocation registries are stored on a blockchain to allow for public checks of privacy-preserving revocation information. Which credentials are accepted in the KYC process may be defined by each bank, depending on its requirements and trust relationships. The combination of eIDAS and DIDs could allow for qualified digital signatures that comply with eIDAS (European Commission, 2019). Credential issuers use institutional agents that are explicitly designed for creating credentials. Besides issuing credentials, these

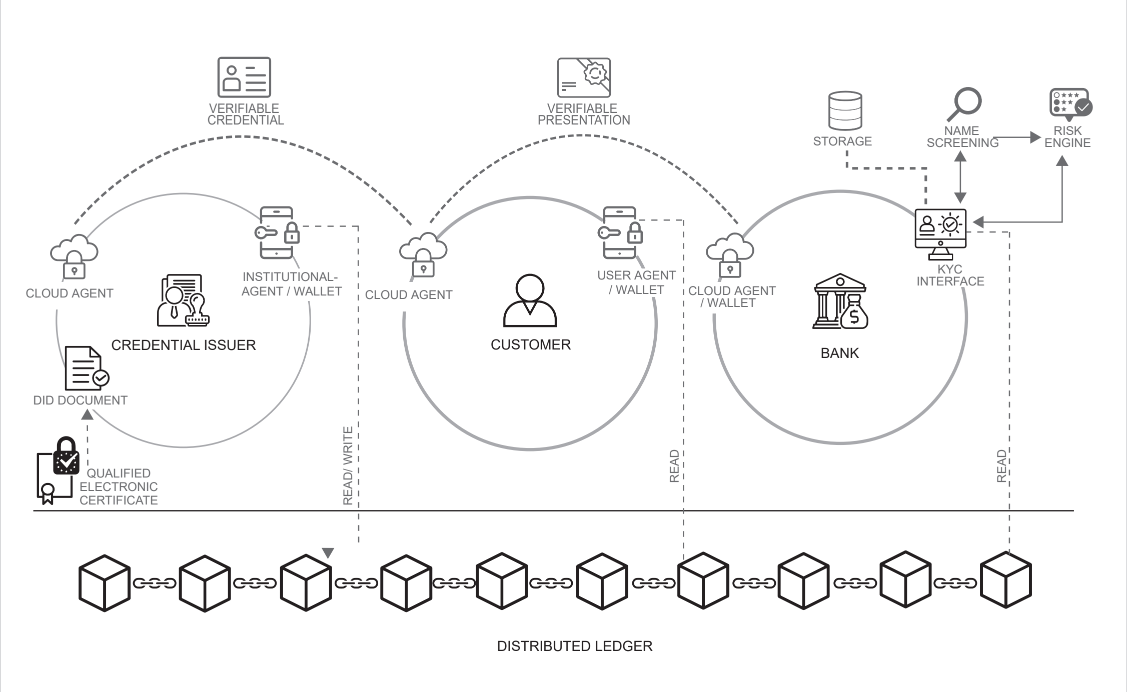


Figure 4: SSI-based KYC architecture (based on European Commission (2019), Moyano and Ross (2017), Reed et al. (2018), Reed et al. (2021), Soltani et al. (2018), and Sporny et al. (2019)).

agents perform identification activities such as checking credentials for integrity and direct communication with the customer that is relevant during and after the KYC process. It also has an interface to name screening services, the bank's risk engine, and customer monitoring. The financial institutions are obliged to store data about customers, for which they use separate storage.

J.5.2 The SSI-based eKYC process

In accordance with the generic procedure of KYC processes, we split the proposed SSI-based eKYC process into three parts: (1) customer identification, data verification, and identity authentication; (2) name screening, risk assessment, and enhanced due diligence; and (3) ongoing monitoring and records keeping. The first part involves three scenarios, depending on the customer's status in the KYC process.

Customer identification, data verification, and identity authentication The KYC process starts with customer onboarding, where three cases can be distinguished. The first case is *completely new onboarding*, where the customer has neither an SSI agent/wallet nor VCs that confirm a completed KYC process. The second case, *fast onboarding*, is possible if the customer already has an SSI agent/wallet with corresponding VCs that attest to the prior completion of a KYC process. Third, we discuss a simplified case we call *new to KYC*, where customers already have an SSI agent or wallet and some VC from other contexts that contain identity-related information trusted by the verifying bank (or that the bank is allowed to trust from a legal perspective), but do not yet have VCs that demonstrate the completion of the KYC process at some institution.

We present the first case, *completely new onboarding*, in a UML sequence diagram (see Figure 5). To enable SSI-based onboarding, as illustrated in (Soltani et al., 2018), banks must conduct a one-time bootstrapping process in which they first store a public DID and an associated DID document in a distributed ledger. This DID document may contain service endpoints of the bank, e.g., for obtaining customer services or conducting the eKYC. The bank will also publish a so-called credential definition, which may be derived from an agreed-on schema/template that contains the attributes that should be attested in a credential, and a revocation registry. All this information is meant to be publicly readable and contains cryptographic information that allows banks (verifiers) to check the validity of VPs that use an associated VC, and customers (holders) to conduct proofs of non-revocation. Thus, these can be stored on the blockchain layer, and no GDPR-related problems are to be expected.

After this initial setup, the bank is ready to perform customer onboarding processes. While (Soltani et al., 2018) presented a (slightly less detailed) sequence diagram for customer onboarding, it involves reading from and writing to the blockchain more often than technically necessary. According to our interview with a co-author of the W3C DID standard, it suffices and is preferable from a privacy perspective to have a peer DID for the customer. We also discuss in detail the implications of the design for the bank and the customer, for instance, related to binding, revocation, and backups. *Completely new onboarding* starts with a bank customer who either visits the bank's website with their smartphone or laptop or physically arrives at a local bank branch to open a new bank account. Since the customer has neither an SSI user wallet nor the necessary KYC credentials, the bank recommends or offers a user wallet and provides the customer with a corresponding download link. Customers can download any digital wallet of their choice that supports the public DID, peer DID, and VC standards. It stores credentials and keys and is secured by a password or biometrics. The bank could further offer an edge agent in encrypted form in the cloud as capability for backup and recovery. The user wallet creates a new DID and some associated keys required for encryption and stores them in the wallet. At the start, the user also creates a so-called link secret, which will later be used to tie different credentials together in a VC and thus provides a means to prevent

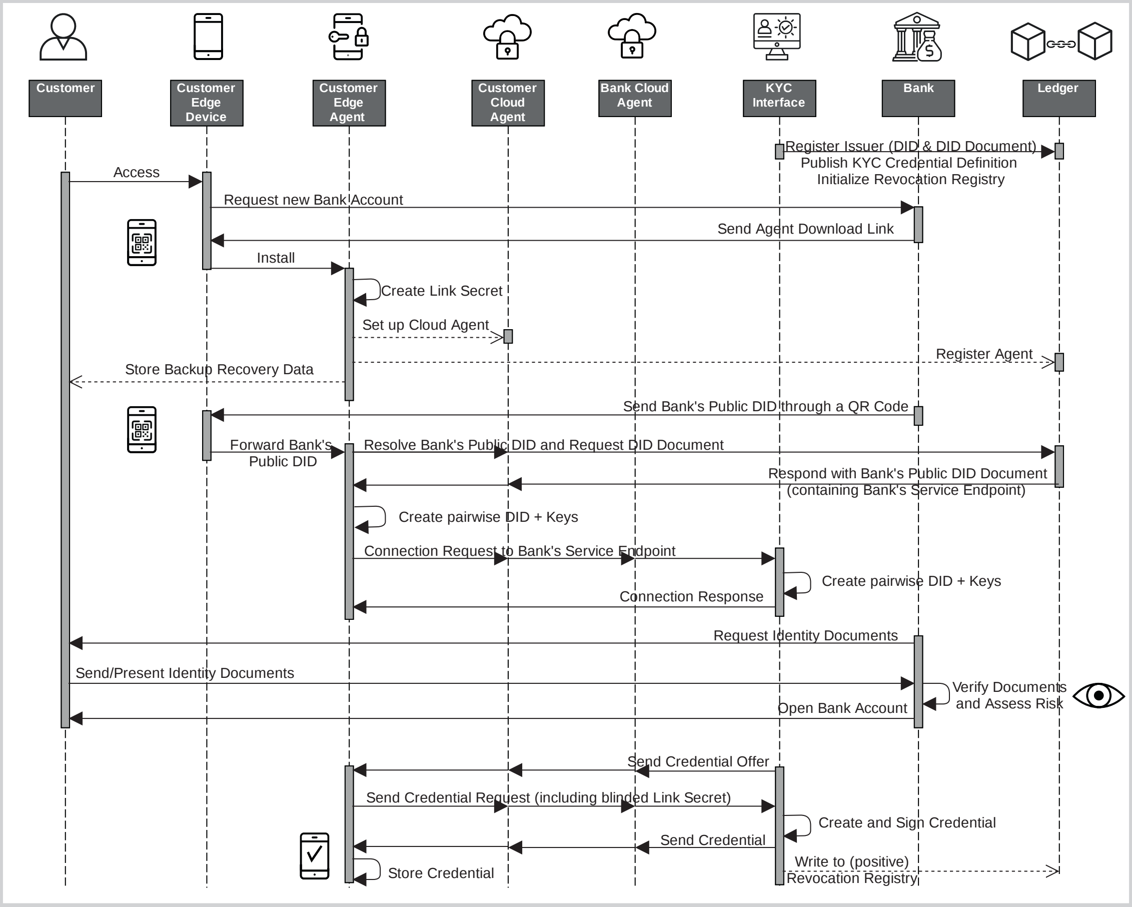

Customer
Customer Edge Device
Customer Edge Agent
Customer Cloud Agent
Bank Cloud Agent
KYC Interface
Bank
Ledger
Register Issuer (DID & DID Document)
Publish KYC Credential Definition
Initialize Revocation Registry
Access
Request new Bank Account
Send Agent Download Link
Install
Create Link Secret
Set up Cloud Agent
Register Agent
Store Backup Recovery Data
Send Bank's Public DID through a QR Code
Forward Bank's Public DID
Resolve Bank's Public DID and Request DID Document
Respond with Bank's Public DID Document (containing Bank's Service Endpoint)
Create pairwise DID + Keys
Connection Request to Bank's Service Endpoint
Create pairwise DID + Keys
Connection Response
Request Identity Documents
Send/Present Identity Documents
Verify Documents and Assess Risk
Open Bank Account
Send Credential Offer
Send Credential Request (including blinded Link Secret)
Create and Sign Credential
Send Credential
Write to (positive) Revocation Registry
Store Credential

Figure 5: UML diagram: Completely new onboarding.

selective credential sharing. However, as long as all credentials contain the customer's name or another strongly binding attribute that needs to be revealed in the VP, it is easy to prove that they belong together also without a link secret.

The customer can now use their newly generated DID to establish an end-to-end-encrypted (secure) connection to the endpoint that the bank offers for the eKYC. The bank could also provide this information by submitting a QR code to the customer (e.g., via e-mail). The customer scans this QR code with their wallet app and thus connects to the bank's public service endpoint. The bank's service behind this endpoint now also creates a new pairwise DID as well as a key pair that the bank will use exclusively in this relationship, and sends a connection request to the customer's service endpoint, its cloud agent, which forwards the connection request to the customer's wallet app. This connection request contains the bank's pairwise DID, the public key used by the bank, and the service endpoint at which the customer can contact the bank, and could also involve a proof that the pairwise DID has in fact been authorized by the bank (e.g., through a VP in which the bank reveals its legal identifier that has been certified by a reputable public institution). In turn, the customer's digital wallet checks the connection's authenticity and creates a pairwise DID and keys for the relationship with the bank. Next, it sends a connection response to the bank's cloud agent/wallet, which forwards it to the bank's KYC interface. Now an end-to-end-encrypted connection exists between the bank and the customer, which can be used to securely exchange messages, public keys, VCs, and VPs. Since the customer does not yet have VCs, the customer's identity must first be verified. The customer sends the necessary analog identity data to the bank, either by traditional means or – if feasible – in scanned form via e-mail or the just-established connection. If the customer opens an account in a bank branch, the documents can also be verified directly there.

After the data have been verified and the customer's identity has been confirmed, the bank can send a credential offer to the customer's edge user agent via the established connection. This credential offer contains a preview of the data that will be attested, the credential issuer information, an expiration date for the VC, and information regarding credential revocation. The customer then accepts the credential offer and sends it to the bank, containing the link secret in blinded form.[2] However, the customer only has to create the link secret once and can later reuse it for their other VCs. The bank includes the blinded link secret in the attributes attested in the VC and sends the VC to the customer. The credential could support selective disclosure. That is, in any VP, the customer can include only the attributes attested by the VC that are necessary for the verifier, and combine claims from different VCs into a VP.

If the issuer wants to support revocation and has bootstrapped a revocation registry, the VC also contains information on how to check its revocation status. The credential issuer can then revoke credentials by updating a revocation registry in the distributed ledger. The bank, for instance, can use this mechanism to invalidate a credential that turns out to be wrongly issued. Notably, it is only through the additional information regarding revocation in the credential that the customer can make sense of the information in the public revocation registry and create a proof of non-revocation within a VP that contains attributes from this VC. Since the credential is never revealed, but only proofs are derived from this, this likely makes public revocation registries compliant with the GDPR.

[2] To be precise, the blinded link secret is a cryptographic commitment, i.e., the hash of the link secret and some one-time random number. Thus, while the blinded form will differ in each credential issuing process, the customer (holder) can still prove that different commitments originate from the same link secret, without revealing the link secret itself, in a zero-knowledge proof (ZKP).

Going beyond (Soltani et al., 2018), we present in detail how the reuse of a KYC process works. This *fast onboarding* process also begins with a bank customer visiting the bank's website or a local bank branch to open a new bank account. The customer states that they already have an SSI user agent and VCs. In the case of opening an account online, the bank sends the customer its bank public DID, for instance by means of a QR code that can be scanned by the customer's wallet app. The channel by which the customer receives this information must be trusted, as the customer does not know the bank's DID in advance. Using an identity infrastructure such as eIDAS, the customer could check whether they are really communicating with the corresponding bank by verifying an eIDAS certification on the bank's public key in the DID document. The customer's user agent can then identify the distributed ledger that stores the DID document associated with the DID and can query this ledger for the DID document. The user agent uses the DID document to identify the bank's eKYC-related service endpoint. The customer's user agent now creates a pairwise DID for this relationship and the corresponding keys and sends a connection request to the bank. The connection request also contains the customer's pairwise DID and the public key used. The bank then creates a pairwise DID and corresponding keys, and sends the customer a connection response, including the pairwise DID and public key.

After establishing the secure connection, the bank sends a proof request for conducting *fast onboarding* KYC. This request contains a random nonce to prevent replay attacks and specifies which data the customer must transmit to the bank, and restrictions on when to accept the VP. This includes a specification of issuers (credential definitions) and schemas that are accepted for the VCs used, and whether there is a need for a proof of non-revocation, including a timestamp of the revocation registry that the customer should refer to in creating this proof if a proof of non-revocation is demanded. The customer's edge agent automatically searches for VCs stored in the customer's digital wallet that match these requirements, updates their local revocation registry through a query if it has not been cached before, and creates a VP that it sends to the bank. The bank can now cryptographically verify the claims, which may also involve reading from revocation registries and other information regarding credential definitions unless sufficiently timely local data from previous queries are cached.

The bank can now cryptographically verify the proof, which involves checks that the digital signatures of issuers were on all attributes involved in the VP, that none of the attested attributes came from a revoked VC, and that all the VCs involved were issued to a commitment of the same, common link secret. After the proof has been verified, the bank account is opened. This whole process can be highly automated and is completed

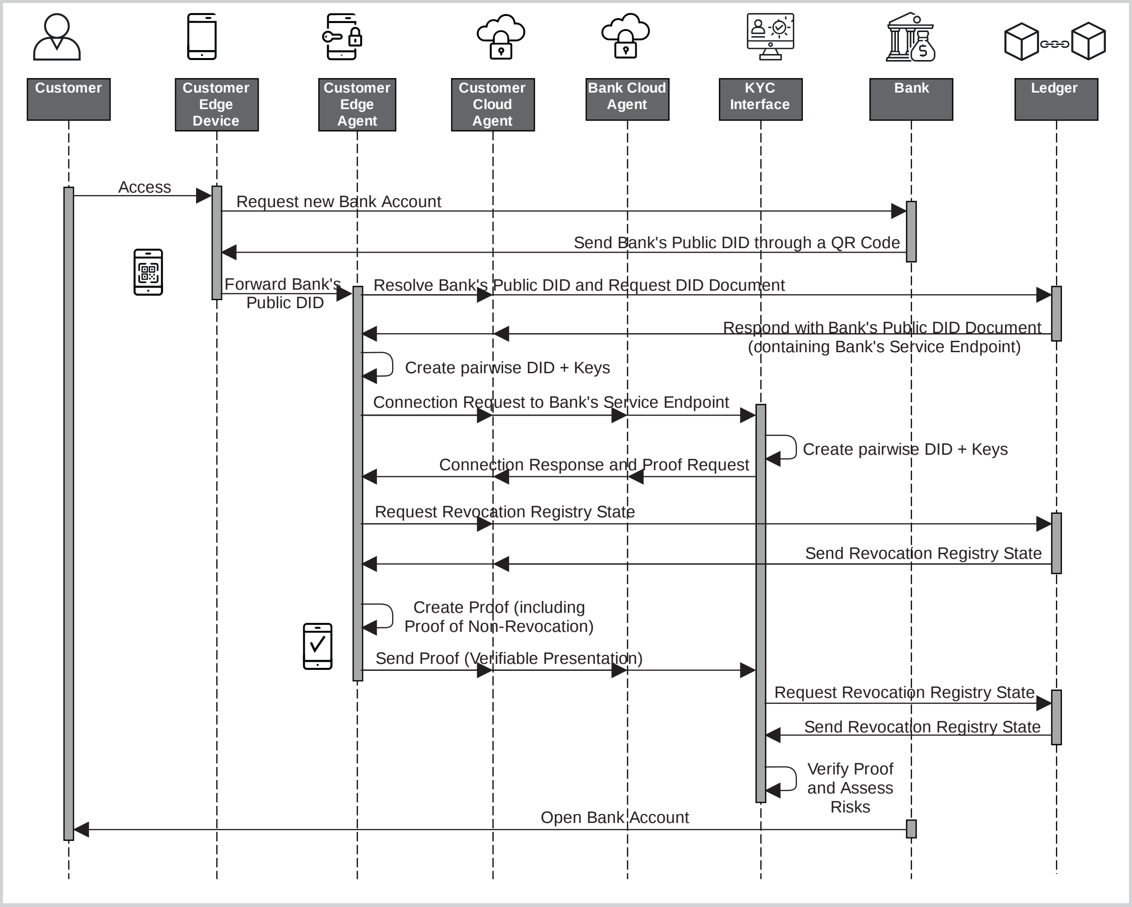

Customer
Customer Edge Device
Customer Edge Agent
Customer Cloud Agent
Bank Cloud Agent
KYC Interface
Bank
Ledger
Access
Request new Bank Account
Send Bank's Public DID through a QR Code
Forward Bank's Public DID
Resolve Bank's Public DID and Request DID Document
Respond with Bank's Public DID Document (containing Bank's Service Endpoint)
Create pairwise DID + Keys
Connection Request to Bank's Service Endpoint
Create pairwise DID + Keys
Connection Response and Proof Request
Request Revocation Registry State
Send Revocation Registry State
Create Proof (including Proof of Non-Revocation)
Send Proof (Verifiable Presentation)
Request Revocation Registry State
Send Revocation Registry State
Verify Proof and Assess Risks
Open Bank Account

Figure 6: UML diagram: Fast onboarding.

in a few seconds. The secure channel established between the bank and the customer based on pairwise DIDs can be used in the future to exchange further documents and to communicate securely and reliably. Based on the exchanged keys, a unique authentication of the involved entities is thereby possible. This is important when dealing with digital identities, since the bank must ensure that it communicates with the same person over time (Jessel et al., 2018).

The third case is *new to KYC*, where the customer already has an SSI user agent and maybe even identity-related VCs, but does not yet have a VC that is accepted by the banks during the KYC process. Thus, *new to KYC* is a combination of *completely new onboarding* (Figure 5) and *fast onboarding* (Figure 6). While the construction of the pairwise DID relationship between the bank and the customer corresponds to the fast onboarding process, the transmission of the analog ID documents and the possibility of getting a KYC credential from the bank corresponds to the process of *completely new onboarding*. However, the customer could first check, through a proof request, whether only a subset of ID documents is necessary because some digital identity proofs are already in their wallet. In addition to the option in which the customer opens an account online, it is also possible to open an account directly in a bank branch by using a QR code to receive the bank's service endpoint and have the analog ID documents checked directly in the bank.

Name screening, risk assessment, and enhanced due diligence After the identity data has been exchanged and cryptographically verified, the name screening service runs in the background of the bank's IT system to check the data against publicly known blacklists regarding terrorism, illegal money laundering activities, politically exposed persons, and negative press. The result of the name screening service is then fed directly into the risk engine, which uses this and other information to classify the customer into a risk class. The risk engine then calculates a risk score and classifies the customer into low, standard, or high risk. Depending on this result, further checks may be necessary before the bank opens the account. Since, in contrast to analog ID documents, VCs are much harder to forge, it suffices to request a minimum amount of information at the start of the relationship. Depending on the risk assessment's result, additional checks may become necessary later. To mitigate risk, the bank can use the previously established secure communication channel to request additional documents and information, such as an income statement or the reason for opening the account. As illustrated, such additional documentation could again be provided in analog form or by using VCs that are already in the customer's wallet – for instance, an income statement issued by an employer that the verifying bank trusts – and deliver an associated VP. Once the customer's verification is successfully completed, the bank can open the account.

Ongoing monitoring Once the account has been opened, the risk engine checks the customer's ongoing transactions during the business relationship, compares these to the expected transaction volume, and checks the transactions for suspicious transaction patterns. Further, the risk engine regularly checks whether the expiration dates contained in the VPs have expired; these may even occasionally trigger a new proof request to the customer through the secure connection to ensure that none of the customer's VCs that were used for KYC have been revoked. The customer then only has to press a confirm button to deliver a new VP. Thus, a manual check of the identity documents is no longer necessary. If it turns out in a periodic refreshment that a customer's VC has in the meantime been revoked (which may just be because of a change of address or a successive re-issuance of an ID card) or that the transaction behavior is abnormal, the risk engine reassesses the risk and proposes measures to mitigate these risks if necessary. The bank can then also request an updated version of the customer's VCs or further documents. This could even be extended to offering the customer an option to automatically send updated versions of their VCs (e.g., if the address on the customer's government-issued identity VC changes) to the bank after the KYC process, so that no more manual activities by the bank and the customer are necessary to keep the data up to date.

Record keeping The SSI concept theoretically allows a bank not to store personal data about its customers at all. The data are solely stored in the customer's digital wallet, and it is very easy to request data when needed and convenient for the customer to provide this information. However, depending on the specific regulations, the banks may be obliged to store their customers' data for a longer period in order to be able to unambiguously determine the person's identity in the event of suspicious or illegal conduct. Therefore, the bank also stores the data in a local database in the redefined KYC process. However, the bank may still manipulate some data. A benefit of ZKP-oriented VCs is that the VP could be made either repudiable or non-repudiable, ensuring tamper-proof documentation where required and supporting customer privacy even in the case of hacks if no auditability is required or sensitive information such as income is involved (Hardman, 2020).

J.6 Evaluation

We now report on a summative, criteria-based evaluation of our proposed framework with interviews with experts (as described in Section J.3), evaluating each of the objectives derived in Section J.4 and their associated requirements in detail (March and Smith, 1995).

Efficiency According to Experts E and I, the SSI-based KYC process presented in the framework has the potential to solve the inefficiencies in the existing KYC process. This can mainly be achieved because the framework involves fully digital cryptographic proofs in the form of VCs. By processing the data entirely digitally (R 1.1), friction in the onboarding process can therefore be reduced for both the customer and the bank (Experts D, H, and I). The need for face-to-face verification, manual data processing, and repeated KYC processes can be eliminated through the use of re-usable VCs combined with revocation registries on the blockchain, thus saving costs for manual and repeated process steps (R 1.2). Expert H also emphasized that updates and periodic confirmation that customers need to provide to banks regarding their data can be significantly reduced through the bilateral and secure communication channel, through which the customer can easily give VPs to the bank. In addition to the potential personnel cost savings, the possibility of authentication with a high level of assurance and the associated reduction of risks can also avoid high penalties for non-compliance with due diligence regulations and standards. However, if there is not yet an existing ecosystem of official identity-related documents, this is only true for the *fast onboarding* process, where a prior eKYC process at another bank or official document issuer has taken place. Standards for KYC credentials can be created and stored on a public blockchain, such that they can be referred to and accepted by a range of institutions (R 1.3). This standardization can be particularly valuable when verification of unknown foreign documents can be avoided (Expert I). Nonetheless, questions regarding governance (e.g., who defines standards) remain open. An additional governance framework is therefore necessary to create clear guidelines for defining which institutions are suitable as credential issuers.

Regulatory compliance Regarding compliance with the MLA, the interviewees did not see particular difficulties in the framework design (R 2.1). The GDPR grants the right to erasure of personal data if the reason for their processing no longer exists. While the de facto interpretation of this regulation remains unclear, it must be assumed that encrypted and hashed personal data also fall under this regulation (Expert C). Further, public DIDs and public keys could be considered as personal data under GDPR, and must therefore be deleted if customers request this (Expert E). Thus, KYC designs that use distributed ledgers to store such data cannot be implemented by banks. In our framework, natural persons only use pairwise DIDs and exchange information bilaterally without writing it to a distributed ledger (Experts E, F, and G). Further GDPR requirements, such as data minimization, are also naturally addressed through VCs' selective disclosure capabilities. The fundamental objectives of our eKYC process are therefore aligned with those of the GDPR (R 2.2). However, a detailed legal assessment remains an avenue for future research.

To effectively use the system, it must also comply with eIDAS regulation (Expert F). The experts noted that they do not see a conflict between eIDAS and the SSI-based eKYC process (R 2.3), and supported the idea of combining the SSI concept and the eIDAS infrastructure (Experts D, E, F, G, and I). Expert G stressed that "these regulations are drivers that will help to adopt SSI, because SSI is an ideal way to implement them." The EU has started building the *eIDAS bridge*, which seeks to make the legally qualified signatures from eIDAS accessible for the VC standard; nonetheless, this implementation has not yet been completed.

Decentralization Our framework for eKYC stores identity-related data in the customer's digital wallet, i.e., on a mobile phone or laptop. Besides the need for banks to store customer information for a certain period – owing to regulatory compliance, rather than for technical reasons – central storage is therefore unnecessary (R 3.1). User agents, whose role is discussed in R 6.3, could be considered for centralized storage. However, these only store data encrypted under user-managed keys. Further, owing to the heavy standardization associated with SSI, it is unlikely that user agents hosted by third parties will encounter the same network effects that have led to centralization for traditional identity providers in federated systems. Thus, the framework counteracts data silos that are highly attractive to hackers, since one can no longer capture many data sets at once (Experts E and F).

The proposed framework also induces no new central parties to the KYC process (R 3.2). Through the use of blockchain, no single entity controls the infrastructure that is involved in checking credential schemas or revocation registries. Expert F mentioned the banks' position of trust toward their customers, and therefore considered the banks to be very suitable providers of cloud agents and wallets. Further, most of the experts support the idea of using banks as potential service providers to backup facilities (Experts E, F, and G).

Trust In our framework, VCs form the basis of KYC documents. The combination of VCs, as an evolution of digital certificates with additional capabilities such as selective disclosure and privacy-preserving revocation mechanisms based on a blockchain, yield a natural digital equivalent of physical KYC documents that customers can fully control and take to other banks. VCs' integrity can be tested by checking the digital signatures' validity, whereby the signing keys of issuing institutions (such as other banks) are publicly available on a blockchain. It is not possible to create fake credentials, because these are not valid without a credential issuer's signature. Further, ownership of credentials can be cryptographically proven (Rannenberg et al., 2015), and binding multiple credentials via a strongly correlating attribute such as the holder's name or biometric properties, or cryptographically through secure hardware, makes credential sharing or selling difficult and unattractive (R 4.3). From the perspective of both banks and regulators, fully digital verification provides a significant advantage over analog documents, since data accuracy is improved and manual errors during data processing can be ruled out (R 4.1) (Experts D, E, and I).

The use of the blockchain infrastructure for storing information on credential issuers (e.g., other banks or government institutions) and revocation registries for VCs provides an infrastructure that allows a bank to verify VCs issued by other banks. Nonetheless, governance mechanisms regarding the legal acceptance of such VCs and other aspects of interbank collaboration required for (R 4.2) still leave some questions open (Experts D, F, G, and I). Such a framework is necessary to clarify which credentials the banks accept and whom they accept as a credential issuer.

Privacy In our framework, users can store and manage their identity data independently, without having to rely on a distinguished third party. Communication is designed to be only bilateral between a credential owner and verifier, and only requires occasional, potentially anonymized read queries to a random node on a public blockchain to update schemas, issuers' signing keys, and revocation registries. This architecture prevents third parties from surreptitiously gaining insights into users' comprehensive data, as is the case with federated identity providers (R 5.1). This is also desirable from a scalability perspective (Expert G). As a result, users can have different digital identities in different contexts and only need to disclose the data required for a specific situation (Experts E and F) in accordance with the *need to know* principle.

In the SSI-based approach, customers have complete control over the data in their wallets, and customers can decide for themselves whom they wish to share data with (Experts D, E, and F). In this context, the experts also mentioned the possibility of selective disclosure through ZKP. The fact that customers no longer have to show all their personal details, but only the relevant data, helps to protect customer privacy through data minimization (R 5.2) (Experts E and F).

The experts also emphasized that a correlation of data is still possible in the absence of public identifiers on the basis of the available rich data sets of banks and other organizations. However, SSI's goal is not anonymity, as is sometimes suggested, but rather the best possible extent of privacy in each scenario. Since KYC procedures seek to build trust, a large amount of personal information must be revealed. In this context, it is important to note that researchers such as Lootsma (2017) have emphasized the possibility of harnessing additional potential through KYC data by, for instance, connecting them to transactional data. While Lootsma (2017) have even raised the question of resulting conflicts with customer privacy, such efforts may also lead to an inherent conflict with SSI principles. Nonetheless, once personally identifiable information has been received in plain text through a bank, it cannot be hindered in connecting it to other data, also in our approach.

User experience The SSI-based eKYC process has the potential to vastly improve the user experience of customers. Much of the current friction, such as entering personal data in an online form, the need to visit a bank, or the need to make a video call with a bank employee for identification process, has been eliminated. Instead, the onboarding process can be carried out on the user's smartphone with just a few steps, for instance by scanning QR codes and accepting invitation links and proof requests through simple interfaces (R 6.1). Because the framework builds on generic and open standards, for which many reference implementations for mobile phones and computer operating systems are available, different user interfaces are realizable (R 6.2). Further, customers have a permanent overview of whom they shared data with (Expert E). A potential problem lies with the customer's full responsibility for data storage and administration (Experts D, E, and G) (R 6.3). Customers must develop an awareness of this so that they realize their responsibility and take appropriate backup and recovery measures to mitigate the consequences of device loss or theft (Experts E and G).

J.7 Discussion

As indicated in Section J.6, our framework can greatly improve KYC processes. In particular, efficiency, trust, and privacy seem to benefit from the blockchain-based SSI architecture. However, many of the improvements do not specifically relate to the KYC case. The trust relationship illustrated in Figure 2 between a holder of ID attributes, an issuer of documents confirming these attributes, and a verifier is present in multiple domains. Thus, our architecture and its related processes reveal insights into the general design of artifacts in the nascent field of blockchain-based SSI, which according to the interviews with the SSI experts may translate to many other areas, where the fear of a centralized service provider has so far prevented a more efficient cross-organizational identity management. To elevate our IT artifact for further theoretical discussion, we derived three DPs that abstract our findings and that seek to guide future research and practice in blockchain-based SSI (Gregor and Hevner, 2013). We analyzed codes from the interviews related to our architecture's technical building blocks (e.g., (distributed) ledger, blockchain, VC, or storage) to identify commonly proposed design patterns and their justification, and we arrived at three generic principles.

Design principle 1: Utilize blockchain only for public data

Our research suggests that the absence of a centralized platform operator in the eKYC process can enable cooperation between banks. The banks do not have to fear that other banks or even a central eKYC utility will receive valuable customer data, which could put them in a disadvantageous position or create new dependencies and lock-in effects. In this context, a distributed ledger is well suited to transparently display public information. On the other hand, owing to their inherent properties – such as transparency, redundancy, and tamper-resistance – blockchains are not suitable for storing personal data (Zhang et al., 2019), even in encrypted form (COVID-19 Credential Initiative, 2021; Finck, 2018). The academic literature often states that credential hashes and peer DIDs also need to be stored on a distributed ledger (Mühle et al., 2018), and initial frameworks for KYC based on SSI (Soltani et al., 2018) have used this approach. However, from a technical perspective and according to the experts, this has no apparent advantages and only carries performance challenges and regulatory risks: Trust in the interaction with a DID is established through a VP, and VCs' tamper resistance is established via the issuer's digital signatures, which need to be trusted anyway. This renders on-chain hashes unnecessary (Toth and Anderson-Priddy, 2019). Further, it must be assumed that legal persons' DIDs fall under the GDPR (Wagner et al., 2018), and for the aforementioned reasons, they should not be stored on a distributed ledger. Thus, distributed ledgers should only be used in a manner comparable to a public key infrastructure (PKI) for VC issuers (Experts E and F) and not for private persons' DIDs and VCs (Experts E, F, and G). By taking most communication off-chain, as Expert G mentioned, the proposed architecture and SSI could help many blockchain use cases to comply with regulation such as the GDPR or eIDAS and could resolve privacy issues (Experts C, D, E, F and G). Regarding performance, the Hyperledger Indy blockchains on which many SSI systems rely can handle only a limited number of write transactions (Sedlmeir et al., 2021a); thus, one should design interactions between stakeholders in a blockchain-based SSI environment bilaterally if possible, and one should read from a blockchain rather than write to it so as to avoid scalability issues.

To abstract and generalize this observation, we propose that by using SSI in processes that require proofs about the possession of certain attributes, organizations should repeatedly request and verify these attributes through bilateral communication channels, instead of storing the required data centrally. As a side effect, this can also help keep data up to date.

Design principle 2: Anticipate an ecosystem of various ledgers

Our initially designed framework was built on the assumption that financial institutions share a single distributed ledger to create and manage digital identities for eKYC. Employing a shared ledger facilitates interoperability on a technical level and concerning governance. However, recent developments in SSI practice (Kuperberg, 2019) and our interview findings indicate that it is more likely that various distributed ledgers for SSI will exist (Experts F and G), similar to the considerable number of today's certificate authorities. Thus, it is important to account for this circumstance and to design blockchain-based SSI solutions for various distributed ledgers to achieve interoperability. This can be achieved through adherence to industry standards, which are currently being developed by organizations such as the W3C (Sporny et al., 2019), as well as by using technical components for interoperability and trust. Universal resolvers – i.e., identifier resolvers working for a multitude of identifiers such as DIDs on different blockchains and maybe also centralized databases (e.g., provided by certificate authorities) – may play an important role in this regard and may also increase trust (Experts D, E, F, and G). While interoperability is technically achievable without major challenges, the existence of various distributed ledgers may induce governance-level challenges that must also be designed through cross-ledger governance (Expert G).

Design principle 3: Enable decentralization at the edge

During the creation of the SSI-based KYC framework, we encountered some challenges regarding SSI-based identity management's user-friendliness. Although managing all identity-relevant data through a single app can boost the straightforward and user-friendly management of identity documents and increase authenticity through hardware-binding or credential-linking, self-managing leads to multiple challenges. For instance, users need support if their devices are lost or stolen. The status quo of central systems must be broken down somewhat here (Wagner et al., 2018), and users must develop an awareness of their responsibility for their data and must learn to store them accordingly (Experts E and F). On the other hand, one must also find the right balance between decentralized and central solutions (Dunphy and Petitcolas, 2018). An example can be the use of cloud storage and cloud agents, which can add value concerning recovery, availability, and security if the agent specializes in this service. On the other hand, these cloud solutions contradict SSI's basic idea of avoiding as many third parties as possible, particularly honey-pots of data, for privacy and security reasons. However, as long as the data are encrypted and the cloud providers cannot access the data, Expert C sees cloud storage a both a viable and an essential element for enabling good user experience. To think decentralization to an end and support the autonomy of end users in blockchain-based SSI applications, SSI-based architectures must ensure that users can store their VCs on an infrastructure of their choice.

Crossing the chasm: How to bring blockchain-based SSI into practice in KYC and beyond

While design artifacts, as the outcome of the DSR process, should have practical impact (Baskerville et al., 2018), bringing the artifacts into practice requires suitable approaches. One key topic mentioned by multiple experts – but that does not relate to technical terms and is therefore not a DP that we can derive based on our codes related to technical building blocks – is the adoption of the SSI-based eKYC. These experts regarded the general adoption of SSI technology in the public and private sectors as a major driving force of the practical implementation of our eKYC solution. Expert D called this a *chicken and egg* problem: Since the technology is still very new, there are few credential providers, so the utility for a user is very low; on the other hand, as long as there are only a few users, there is also no major incentive for organizations to act as credential issuers. The more credentials users have, the better such a system can be used (Expert D). Network effects can help bridge the gap between early adopters and the widespread use of the technology (Moore and McKenna, 1999). In particular, banks can contribute to this by offering the technology in the *isolated* KYC use case, issuing VCs to contribute to its spread in the mass market. The more that banks accept these credentials, the more attractive the system becomes to customers. In turn, higher usage by customers leads to more incentive for other organizations to accept VCs. As mentioned, the cooperation of the banks and the creation of shared standards are crucial if this adoption is to become possible.

Because users have full control over their data, SSI must address the GDPR's general requirements, such as privacy by design, portability, the right to erasure, transparency, purpose limitation, data minimization, accuracy, storage limitation, and information integrity. Users can get a permanent overview of whom they shared what data with (Expert E), and these records can help to better implement the right to erasure. In turn, this may lead to a better adoption of the technology. In fact, the GDPR's strict requirements, which were often criticized for impeding blockchain-related innovation in Europe, may ultimately have turned out to boost innovation, so blockchain's benefits for interoperability can be used for the purposes highlighted in DP 1 while avoiding its well-known privacy- and scalability-related challenges.

The interplay between SSI and regulation uncovers many other interesting dimensions. For instance, our interviewees suggested that eIDAS regulation can facilitate SSI while SSI can help make the eIDAS infrastructure, which so far has been used only moderately, more practicable and valuated (Lyons et al., 2019). On the other hand, we saw that SSI technically allows for even more privacy than what is required by regulation. However, the MLA requires that banks store customer data for five years, creating tension between data protection regulation and the objectives of user control and the prevention of data silos. Thus, SSI may even lead to new discussions on where to set the sweet spot between market integrity and privacy.

J.8 Conclusion

In this article, we sought to build a framework to improve on the current shortcomings in the KYC process through an end-to-end digital process that leverages blockchain-based SSI. Research on SSI is still in its infancy, and little has been published on the design of applications for SSI. Soltani et al. (2018) were the first to explore this topic in the context of KYC, covering the onboarding process and technically evaluating their solution. Building on this valuable work, we extended the scope and emphasized banks' requirements. We used a DSR approach based on Peffers et al. (2007), designing and evaluating a framework for KYC processes built on blockchain-based SSI, including a generic architecture and process design. Since we face a low solution maturity in the innovative field of blockchain and SSI, and high application domain maturity in the domain of KYC, we provided an *improvement* in the context of DSR (Gregor and Hevner, 2013). Our evaluation suggests that our design can significantly contribute to a more efficient KYC process that also addresses the other requirements of stakeholders. Thus, we are confident that we have accomplished our research objective.

Besides the conceptualized and evaluated architecture and set of processes (Gregor and Hevner, 2013) for the KYC case, we made three primary contributions to the academic body of knowledge. First, our examination revealed the challenges of using DLT for the exchange of personal data generally and particularly for digital identity management systems. We also showed how these problems can be solved by using SSI on top of the blockchain layer, thereby leveraging the advantages typically associated with blockchain technology while avoiding its well-known issues with scalability and privacy. Second, we revealed the implications of designing SSI-based solutions built on blockchain in the context of KYC by deriving three DPs, which allowed us to elevate our IT artifact for more abstract and generalizable theoretical discussion (Gregor and Hevner, 2013). Third, we offered suggestions for relevant future research on blockchain and SSI, enabling researchers to base their work on our results and thus generate additional knowledge (vom Brocke et al., 2020).

DSR should also inform practice to advance a specific domain through IT (Gregor and Hevner, 2013). Our conception and evaluation of the SSI-based KYC framework will provide practitioners with valuable insights regarding design choices, DLT's role, the intricacies of regulation, and related challenges and opportunities for banks and customers. Our results indicate that SSI-based eKYC processes can reduce cost and time expenditures and contribute to better user experiences and increased security during the KYC process. We demonstrated how the use of SSI can positively impact the different onboarding processes and their interplays with an existing SSI ecosystem. However, we illustrated that there are further conceptual challenges to be solved before SSI is used in real systems and settings, especially regarding the necessary governance frameworks and a more detailed regulatory analysis. While our research suggests synergies between SSI and regulation, challenges remain, especially to establish a general SSI-based ecosystem and to make SSI as user-friendly as possible without sacrificing privacy and security.

Like most research, our study has limitations. Our framework has not yet been used in practice and therefore lacks an evaluation in a real-world setting. However, by applying a rigorous research design and obtaining practitioner feedback, we sought to address this shortcoming. Further, although we described the necessary and central elements of an additionally required governance framework, its concrete implementation remains open. In particular, details regarding the cooperation of banks in the KYC process, the creation of shared standards, and the responsible parties for operating the blockchain in the case of using a permissioned network must be clarified. This opens various promising avenues for future research. In particular, the KYC process is often relevant not only in the banking environment but also in other domains such as insurance, and further objectives may be necessary to address this aspect. Further, although our interviews with experts confirmed Moyano and Ross's (2017) findings that there is sufficient trust between banks that collaboration on eKYC is possible despite competition on a suitable IT infrastructure, this only represents the perspective of practitioners and researchers in central and northern European countries. Nonetheless, there is also a promising development that may make SSI-based KYC and our findings considerably more far-reaching and applicable in contexts in which this trust is missing: In emerging SSI ecosystems in North America and Europe, governments are starting to explore the impacts of issuing certificates such as driver's licenses and ID cards in the form of VCs that can be leveraged by the public and private sector. Besides increasing the efficiency of digital identification and authentication, a digital ID is expected to contribute to substantial reductions in financial crime (Financial Times, 2021). In this context, an experimental clause was recently adopted in Germany's parliament that explicitly allows banks to perform KYC based on an ID card in the form of a VC (Association of German Banks, 2021; Deutscher Bundestag, 2021).

We are confident that we have derived guidelines and DPs that generalize to these promising developments and to other sectors, and that also provide guidance on how to make blockchain-based SSI compatible with the needs of businesses and regulatory restrictions. More efficient, reusable processes, specifically relating to identity management, are needed in both business and the public sector, but the risks associated with central providers have so far prevented general services from providing these capabilities. Blockchain-based SSI can address the need for a general service for data that is non-competitive, since the same data are needed and used by all, and yet comply with customer data privacy expectations and regulations. On the other hand, our DP of minimal involvement of the blockchain, specifically not storing natural persons' DIDs or even VCs on a blockchain, translates to applications of SSI generally, and we are eager to see more use cases being built on this technology stack. Given current efforts by the Verifiable Organizations Network in Canada, the ambitious goal of having 10 different pilots that leverage blockchain-based SSI by the end of 2021 in Germany, and several SSI smartphone wallets that are already available, we are confident that this time, the promises of blockchain-technology to revolutionize digital ID management can be fulfilled, although blockchain's role will be much more restricted than what early research suggested.

References

Allen, C. (2016). *The path to self-sovereign identity*. Life with Alacrity. URL: http://www.lifewithalacrity.com/2016/04/the-path-to-self-sovereign-identity.html.

Arasa, R. and L. Ottichilo (2015). "Determinants of know your customer (KYC) compliance among commercial banks in Kenya". In: *Journal of Economics and Behavioral Studies* 7 (2), pp. 162–175. DOI: 10.22610/jebs.v7i2(J).574.

Arner, D. W., J. N. Barberis, and R. P. Buckley (2016). *The emergence of RegTech 2.0: From know your customer to know your data*. URL: https://ssrn.com/abstract=3044280.

Arner, D. W., D. A. Zetzsche, R. P. Buckley, and J. N. Barberis (2019). "The identity challenge in finance: From analogue identity to digitized identification to digital KYC utilities". In: *European Business Organization Law Review* 20 (1), pp. 55–80. DOI: 10.1007/s40804-019-00135-1.

Association of German Banks (2021). *Digital identities – steps on the path to an ID ecosystem*. URL: https://en.bankenverband.de/newsroom/comments/digital-identities-steps-path-id-ecosystem/#2.

Avellaneda, O., A. Bachmann, A. Barbir, J. Brenan, P. Dingle, K. H. Duffy, E. Maler, D. Reed, and M. Sporny (2019). "Decentralized identity: Where did it come from and where is it going?" In: *IEEE Communications Standards Magazine* 3 (4), pp. 10–13. DOI: 10.1109/mcomstd.2019.9031542.

Baskerville, R., A. Baiyere, S. Gregor, A. Hevner, and M. Rossi (2018). "Design science research contributions: Finding a balance between artifact and theory". In: *Journal of the Association for Information Systems* 19 (5), pp. 358–376. DOI: 10.17705/1jais.00495.

Beck, R., S. Weber, and R. W. Gregory (2013). "Theory-generating design science research". In: *Information Systems Frontiers* 15 (4), pp. 637–651. DOI: 10.1007/s10796-012-9342-4.

Biryukov, A., D. Khovratovich, and S. Tikhomirov (2018). "Privacy-preserving KYC on Ethereum". In: *1st Blockchain Workshop*. ERCIM. DOI: 10.18420/blockchain2018_09.

Butijn, B.-J., D. A. Tamburri, and W.-J. van den Heuvel (2020). "Blockchains: A systematic multivocal literature review". In: *ACM Computing Surveys* 53 (3). DOI: 10.1145/3369052.

Camenisch, J. and A. Lysyanskaya (2001). "An efficient system for non-transferable anonymous credentials with optional anonymity revocation". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, pp. 93–118. DOI: 10.1007/3-540-44987-6_7.

Casino, F., T. K. Dasaklis, and C. Patsakis (2019). "A systematic literature review of blockchain-based applications: Current status, classification and open issues". In: *Telematics and Informatics* 36, pp. 55–81. DOI: 10.1016/j.tele.2018.11.006.

Charmaz, K. (2006). *Constructing grounded theory: A practical guide through qualitative analysis*. SAGE.

Christie, R. (2018). "Setting a standard path forward for KYC". In: *Journal of Financial Transformation* 47, pp. 155–164.

Clauß, S. and M. Köhntopp (2001). "Identity management and its support of multilateral security". In: *Computer Networks* 37 (2), pp. 205–219. DOI: 10.1016/S1389-1286(01)00217-1.

Corbin, J. and A. Strauss (2008). *Basics of qualitative research: Techniques and procedures for developing grounded theory*. SAGE.

COVID-19 Credential Initiative (2021). *Global implementation forum*. URL: https://docs.google.com/document/d/1dbWvs1m8uziTsbhUQv_nPofTXAyDSkxI5CZtoo1SlRY.

Davie, M., D. Gisolfi, D. Hardman, J. Jordan, D. O'Donnell, and D. Reed (2019). "The trust over IP stack". In: *IEEE Communications Standards Magazine* 3 (4), pp. 46–51. DOI: 10.1109/mcomstd.001.1900029.

Deutscher Bundestag (2021). *Drucksache 19/30443*. URL: https://dserver.bundestag.de/btd/19/304/1930443.pdf.

Dhamija, R. and L. Dusseault (2008). "The seven flaws of identity management: Usability and security challenges". In: *IEEE Security & Privacy* 6 (2), pp. 24–29. DOI: 10.1109/msp.2008.49.

Dunphy, P. and F. A. P. Petitcolas (2018). "A first look at identity management schemes on the blockchain". In: *IEEE Security & Privacy* 16 (4), pp. 20–29. DOI: 10.1109/msp.2018.3111247.

El Maliki, T. and J.-M. Seigneur (2007). "A survey of user-centric identity management technologies". In: *The International Conference on Emerging Security Information, Systems, and Technologies*, pp. 12–17. DOI: 10.1109/secureware.2007.4385303.

European Commission (2019). *eIDAS supported self-sovereign identity*. URL: https://ec.europa.eu/futurium/en/system/files/ged/eidas_supported_ssi_may_2019_0.pdf.

FATF (2004). *FATF 40 recommendations*. URL: https://www.fatf-gafi.org/media/fatf/documents/FATF%20Standards%20-%2040%20Recommendations%20rc.pdf.

Financial Times (2021). *81 % of financial services firms agree digital ID would improve financial crime prevention*. URL: https://thefintechtimes.com/81-of-financial-services-firms-agree-digital-id-would-improve-financial-crime-prevention/.

Finck, M. (2018). "Blockchains and data protection in the European Union". In: *European Data Protection Law Review* 4 (1), pp. 17–35. DOI: 10.21552/edpl/2018/1/6.

Fridgen, G., S. Radszuwill, N. Urbach, and L. Utz (2018). "Cross-organizational workflow management using blockchain technology – towards applicability, auditability, and automation". In: *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp. 3507–3516. DOI: 10.24251/hicss.2018.444.

Glaser, F. (2017). “Pervasive decentralisation of digital infrastructures: A framework for blockchain enabled system and use case analysis”. In: *Proceedings of the 50th Hawaii International Conference on System Sciences*, pp. 1543–1552. DOI: 10.24251/hicss.2017.186.

Gregor, S. and A. R. Hevner (2013). “Positioning and presenting design science research for maximum impact”. In: *MIS Quarterly* 37 (2), pp. 337–355. DOI: 10.25300/misq/2013/37.2.01.

Guggenberger, T., A. Schweizer, and N. Urbach (2020). “Improving interorganizational information sharing for vendor managed inventory: Toward a decentralized information hub using blockchain technology”. In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1074–1085. DOI: 10.1109/tem.2020.2978628.

Hardman, D. (2020). *No paradox here: ZKPs deliver savvy trust.* Evernym. URL: https://www.evernym.com/blog/no-paradox-here-zkps-deliver-savvy-trust/.

Hardman, D., L. Harchandani, A. Othman, and J. Callahan (2019). “Using biometrics to fight credential fraud”. In: *IEEE Communications Standards Magazine* 3 (4), pp. 39–45. DOI: 10.1109/mcomstd.001.1900033.

Hevner, A. and S. Gregor (2020). “Envisioning entrepreneurship and digital innovation through a design science research lens: A matrix approach”. In: *Information & Management*, p. 103350. DOI: 10.1016/j.im.2020.103350.

Hevner, A., S. T. March, J. Park, S. Ram, et al. (2004). “Design science research in information systems”. In: *MIS Quarterly* 28 (1), pp. 75–105. DOI: 10.2307/25148625.

Hughes, P. L. (1988). “The ‘need to know’ principle of computer security”. In: *Computer Law & Security Review* 3 (5), pp. 29–30. DOI: 10.1016/0267-3649(88)90114-8.

Jessel, B., K. Lowmaster, N. Hughes, et al. (2018). “Digital identity: The foundation for trusted transactions in financial services”. In: *Journal of Financial Transformation* 47, pp. 143–150. URL: https://ideas.repec.org/a/ris/jofitr/1607.html.

Jones, D. and S. Gregor (2007). “The anatomy of a design theory”. In: *Journal of the Association for Information Systems* 8 (5), pp. 312–335. DOI: 10.17705/1jais.00129.

Kallio, H., A.-M. Pietilä, M. Johnson, and M. Kangasniemi (2016). “Systematic methodological review: Developing a framework for a qualitative semi-structured interview guide”. In: *Journal of Advanced Nursing* 72 (12), pp. 2954–2965. DOI: 10.1111/jan.13031.

Kokolakis, S. (2017). “Privacy attitudes and privacy behaviour: A review of current research on the privacy paradox phenomenon”. In: *Computers & Security* 64, pp. 122–134. DOI: 10.1016/j.cose.2015.07.002.

Kolb, J., M. AbdelBaky, R. H. Katz, and D. E. Culler (2020). "Core concepts, challenges, and future directions in blockchain: A centralized tutorial". In: *ACM Computing Surveys* 53 (1). DOI: 10.1145/3366370.

Kuperberg, M. (2019). "Blockchain-based identity management: A survey from the enterprise and ecosystem perspective". In: *IEEE Transactions on Engineering Management* 67 (4), pp. 1008–1027. DOI: 10.1109/tem.2019.2926471.

Ledger Insights (2020). *Self-sovereign identity successfully trialed for KYC in UK regulatory sandbox*. URL: https://www.ledgerinsights.com/self-sovereign-identity-successfully-trialed-for-kyc-in-uk-regulatory-sandbox/.

Lim, S. Y., P. T. Fotsing, A. Almasri, O. Musa, M. L. M. Kiah, T. F. Ang, and R. Ismail (2018). "Blockchain technology the identity management and authentication service disruptor: A survey". In: *International Journal on Advanced Science, Engineering and Information Technology* 8 (4-2), pp. 1735–1745. DOI: 10.18517/ijaseit.8.4-2.6838.

Liu, Y., Q. Lu, H.-Y. Paik, X. Xu, S. Chen, and L. Zhu (2020). "Design pattern as a service for blockchain-based self-sovereign identity". In: *IEEE Software* 37 (5), pp. 30–36. DOI: 10.1109/ms.2020.2992783.

Lootsma, Y. (2017). "Blockchain as the newest regtech application – The opportunity to reduce the burden of KYC for financial institutions". In: *Banking & Financial Services Policy Report* 36 (8), pp. 16–21.

Lyons, T., L. Courcelas, and K. Timsit (2019). *Blockchain and digital identity*. European Union Blockchain Observatory and Forum. URL: https://www.eublockchainforum.eu/sites/default/files/report_identity_v0.9.4.pdf.

Maler, E. and D. Reed (2008). "The Venn of identity: Options and issues in federated identity management". In: *IEEE Security & Privacy* 6 (2), pp. 16–23. DOI: 10.1109/msp.2008.50.

March, S. T. and G. F. Smith (1995). "Design and natural science research on information technology". In: *Decision Support Systems* 15 (4), pp. 251–266. DOI: 10.1016/0167-9236(94)00041-2.

Moor, J. H. (1997). "Towards a theory of privacy in the information age". In: *ACM SIGCAS Computers and Society* 27 (3), pp. 27–32. DOI: 10.1145/270858.270866.

Moore, G. A. and R. McKenna (1999). *Crossing the chasm*. Harper Business Essentials.

Morse, J. M. (1991). "Strategies for sampling". In: *Qualitative nursing research: A contemporary dialogue*. Chap. 8, pp. 127–145.

Moyano, J. P. and O. Ross (2017). "KYC optimization using distributed ledger technology". In: *Business & Information Systems Engineering* 59 (6), pp. 411–423. DOI: 10.1007/s12599-017-0504-2.

Mugarura, N. (2014). “Customer due diligence (CDD) mandate and the propensity of its application as a global AML paradigm”. In: *Journal of Money Laundering Control* 17 (1), pp. 75–96. DOI: 10.1108/JMLC-07-2013-0024.

Mühle, A., A. Grüner, T. Gayvoronskaya, and C. Meinel (2018). “A survey on essential components of a self-sovereign identity”. In: *Computer Science Review* 30, pp. 80–86. DOI: 10.1016/j.cosrev.2018.10.002.

Myers, M. D. and M. Newman (2007). “The qualitative interview in IS research: Examining the craft”. In: *Information and Organization* 17 (1), pp. 2–26. DOI: 10.1016/j.infoandorg.2006.11.001.

Norvill, R., M. Steichen, W. M. Shbair, and R. State (2019). “Blockchain for the simplification and automation of KYC result sharing”. In: *International Conference on Blockchain and Cryptocurrency*. IEEE, pp. 9–10. DOI: 10.1109/bloc.2019.8751480.

Ostern, N. K. and J. Riedel (2020). “Know-your-customer (KYC) requirements for initial coin offerings”. In: *Business & Information Systems Engineering* 63, pp. 551–567. DOI: 10.1007/s12599-020-00677-6.

Peffers, K., T. Tuunanen, M. A. Rothenberger, and S. Chatterjee (2007). “A design science research methodology for information systems research”. In: *Journal of Management Information Systems* 24 (3), pp. 45–77. DOI: 10.2753/mis0742-1222240302.

Perlman, L. and N. Gurung (2019). *Focus note: The use of eKYC for customer identity and verification and AML*. URL: https://ssrn.com/abstract=3370665.

Rajput, A. and K. Gopinath (2017). “Towards a more secure Aadhaar”. In: *International Conference on Information Systems Security*. Springer, pp. 283–300. DOI: 10.1007/978-3-319-72598-7_17.

Rannenberg, K., J. Camenisch, and A. Sabouri (2015). *Attribute-based credentials for trust – Identity in the information society*. Springer. DOI: 10.1007/978-3-319-14439-9.

Reed, D., J. Law, D. Hardman, and M. Lodder (2018). *DKMS (decentralized key management system) design and architecture v3*. URL: https://github.com/hyperledger/indy-sdk/blob/677a0439487a1b7ce64c2e62671ed3e0079cc11f/doc/design/005-dkms/DKMS%20Design%20and%20Architecture%20V3.md.

Reed, D., M. Sporny, D. Longley, C. Allen, R. Grant, M. Sabadello, and J. Holt (2021). *Decentralized identifiers (DIDs) v1.0: Core architecture, data model, and representations*. Ed. by D. Reed, M. Sporny, and M. Sabadello. World Wide Web Consortium. URL: https://www.w3.org/TR/did-core/.

Reinecke, K. and A. Bernstein (2013). “Knowing what a user likes: A design science approach to interfaces that automatically adapt to culture”. In: *MIS Quarterly* 37 (2), pp. 427–453. DOI: 10.25300/misq/2013/37.2.06.

Rieger, A., F. Guggenmos, J. Lockl, G. Fridgen, and N. Urbach (2019). “Building a blockchain application that complies with the EU general data protection regulation”. In: *MIS Quarterly Executive* 18 (4), pp. 263–279. DOI: 10.17705/2msqe.00020.

Rossi, M., C. Mueller-Bloch, J. B. Thatcher, and R. Beck (2019). “Blockchain research in information systems: Current trends and an inclusive future research agenda”. In: *Journal of the Association for Information Systems*, pp. 1388–1403. DOI: 10.17705/1jais.00571.

Ruce, P. J. (2011). “Anti-money laundering: The challenges of know your customer legislation for private bankers and the hidden benefits for relationship management (the bright side of knowing your customer)”. In: *The Banking Law Journal* 128 (6), pp. 548–564. URL: https://heinonline.org/HOL/P?h=hein.journals/blj128&i=560.

Saldaña, J. (2015). *The coding manual for qualitative researchers*. SAGE.

Schweizer, A., V. Schlatt, N. Urbach, and G. Fridgen (2017). “Unchaining social businesses-blockchain as the basic technology of a crowdlending platform”. In: *Proceedings of the 37th International Conference on Information Systems*. AIS. URL: https://aisel.aisnet.org/icis2017/TransformingSociety/Presentations/8/.

Sedlmeir, J., P. Ross, A. Luckow, J. Lockl, D. Miehle, and G. Fridgen (2021a). “The DLPS: A new framework for benchmarking blockchains”. In: *Proceedings of the 54th Hawaii International Conference on System Sciences*, pp. 6855–6864. DOI: 10.24251/hicss.2021.822.

Sedlmeir, J., R. Smethurst, A. Rieger, and G. Fridgen (2021b). “Digital identities and verifiable credentials”. In: *Business & Information Systems Engineering* 63 (5), pp. 603–613. DOI: 10.1007/s12599-021-00722-y.

Soltani, R., U. T. Nguyen, and A. An (2018). “A new approach to client onboarding using self-sovereign identity and distributed ledger”. In: *International Conference on Internet of Things and Green Computing and Communications and Cyber, Physical and Social Computing and Smart Data*. IEEE, pp. 1129–1136. DOI: 10.1109/cybermatics_2018.2018.00205.

Sonnenberg, C. and J. vom Brocke (2012). “Evaluations in the science of the artificial – reconsidering the build-evaluate pattern in design science research”. In: *International Conference on Design Science Research in Information Systems*. Springer, pp. 381–397. DOI: 10.1007/978-3-642-29863-9_28.

Sporny, M., D. Longley, and D. Chadwick (2019). *Verifiable credentials data model 1.0: Expressing verifiable information on the Web*. Ed. by M. Sporny, G. Noble, D. Longley, D. C. Burnett, and B. Zundel. World Wide Web Consortium. URL: https://www.w3.org/TR/vc-data-model.

Swinhoe, D. (2020). *The 15 biggest data breaches of the 21st century*. IDG Communications. URL: https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html.

The Economist (2016). *Indian business prepares to tap into Aadhaar, a state-owned fingerprint-identification system*. URL: https://www.economist.com/business/2016/12/24/indian-business-prepares-to-tap-into-aadhaar-a-state-owned-fingerprint-identification-system.

Thomson Reuters (2016). *Know your customer surveys reveal escalating costs and complexity*. URL: https://www.thomsonreuters.com/en/press-releases/2016/may/thomson-reuters-2016-know-your-customer-surveys.html.

Toth, K. C. and A. Anderson-Priddy (2019). “Self-sovereign digital identity: A paradigm shift for identity”. In: *IEEE Security & Privacy* 17 (3), pp. 17–27. DOI: 10.1109/msec.2018.2888782.

Trust over IP Foundation (2020). *Introducing the Trust over IP Foundation*. URL: https://trustoverip.org/wp-content/uploads/sites/98/2020/05/toip_introduction_050520.pdf.

Venable, J., J. Pries-Heje, and R. Baskerville (2012). “A comprehensive framework for evaluation in design science research”. In: *International Conference on Design Science Research in Information Systems*. Springer, pp. 423–438. DOI: 10.1007/978-3-642-29863-9_31.

Venable, J., J. Pries-Heje, and R. Baskerville (2016). “FEDS: A framework for evaluation in design science research”. In: *European Journal of Information Systems* 25 (1), pp. 77–89. DOI: 10.1057/ejis.2014.36.

vom Brocke, J., R. Winter, A. Hevner, and A. Maedche (2020). “Special issue editorial – Accumulation and evolution of design knowledge in design science research: A journey through time and space”. In: *Journal of the Association for Information Systems* 21 (3), pp. 520–544. DOI: 10.17705/1jais.00611.

Wagner, K., B. Némethi, E. Renieris, P. Lang, E. Brunet, and E. Holst (2018). *Self-sovereign identity: A position paper on blockchain enabled identity and the road ahead*. Identity Working Group of the German Blockchain Association. URL: https://jolocom.io/wp-content/uploads/2018/10/Self-sovereign-Identity-%5C_-Blockchain-Bundesverband-2018.pdf.

Zavolokina, L., R. Ziolkowski, I. Bauer, and G. Schwabe (2020). "Management, governance and value creation in a blockchain consortium". In: *MIS Quarterly Executive* 19 (1). DOI: 10.17705/2msqe.00022.

Zetzsche, D. A., R. P. Buckley, and D. W. Arner (2018). "Digital ID and AML/CDD/KYC utilities for financial inclusion, integrity and competition". In: *Journal of Economic Transformation* 47, pp. 133–142. URL: https://www.capco.com/-/media/CapcoMedia/Capco-2023/Capco-Institute/Journal-47/CapcoJournal47ZetzschePrintv11.ashx.

Zhang, R., R. Xue, and L. Liu (2019). "Security and privacy on blockchain". In: *ACM Computing Surveys* 52 (3). DOI: 10.1145/3316481.